# PENTESTING PARA EL PORTAL WEB DE LA USPEC, APOYADO EN EL PROYECTO DE SEGURIDAD OWASP

JHONNIER YESID ZÚÑIGA MOSQUERA

UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA (UNAD)
ESCUELA DE CIENCIAS BÁSICAS, TECNOLOGÍA E INGENIERÍA
ESPECIALIZACIÓN EN SEGURIDAD INFORMÁTICA
CCAV QUIBDÓ
2018

# PENTESTING PARA EL PORTAL WEB DE LA USPEC, APOYADO EN EL PROYECTO DE SEGURIDAD OWASP

JHONNIER YESID ZÚÑIGA MOSQUERA

Trabajo de grado para optar el título:
Especialista en seguridad informática

Director del proyecto
EDGAR ALONSO BOJACA GARAVITO

UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA (UNAD)
ESCUELA DE CIENCIAS BÁSICAS, TECNOLOGÍA E INGENIERÍA
ESPECIALIZACIÓN EN SEGURIDAD INFORMÁTICA
CCAV QUIBDÓ
2018

Nota de aceptación

_____

_____

_____

_____

_____

_____

_____

_____

Firma del presidente del jurado

_____

Firma del jurado

_____

Firma del jurado

Bogotá D.C, 24 de Octubre de 2018

## DEDICATORIA

*Dedico este proyecto el creador y todo poderoso DIOS por iluminar el camino y a mi familia, las Cuales han brindado apoyo y confianza siendo Parte fundamental para el progreso en la Formación académica.*

## AGRADECIMIENTOS

La Universidad Nacional Abierta y a Distancia UNAD, por brindar la posibilidad de crecer académicamente, ayudando en mi crecimiento personal y profesional para poder contribuir en con un conocimiento fructífero en la seguridad de la información. Al Ingeniero Salomón González, de igual manera al ingeniero Edgar Alonso Bajaca Garavito por ser los formadores con gran sabiduría que han plasmado directrices que llevan a cosechar los frutos en cada una de sus apreciaciones, su gran inteligencia y dedicación en su labor.

A los compañeros que en el trascurso de la especialización contribuyeron aportando su granito de arena con sus ideales que ayudaron a fortalecer y afianzar estrategias para el mejoramiento de la seguridad en las organizaciones que estemos ejerciendo nuestra labor profesional.

## TABLA DE CONTENIDO

**Pág.**

## TABLA DE ILUSTRACIONES

Pag.

## LISTA DE TABLAS

**Pág.**

## LISTA DE ANEXOS

**Pág.**

# INTRODUCCIÓN

Todas las empresas y/o instituciones hoy día deben asegurar que su portal web cumpla con estándares de seguridad que permitan, fortificar sus sistemas informáticos, y de la misma manera garantizar la seguridad en los procesos gestionados por la empresa a través del portal web. La seguridad de la información debe mirarse de otra manera pues no debemos esperar incidentes para reaccionar sino estar precavidos y resguardar los activos de las empresas. Es importante para la seguridad de las organizaciones que se estudien los casos de seguridad y se analicen mediante metodologías y estrategias que garanticen un fructífero control en el manejo de sus activos informáticos mediante el proyecto OWASP.

En este sentido el proyecto busca hacer un diagnóstico del estado del portal web de la unidad de servicios penitenciarios y carcelarios USPEC. Mediante pruebas de penetración y escaneos que permitan identificar las vulnerabilidades amenazas y riesgos de seguridad informática que pueden afectar el portal web y sus activos informáticos mediante pruebas aplicadas al portal para proponer políticas que puedan ser aplicadas y mitigar problemas de seguridad.

# 2 DESCRIPCIÓN DEL PROBLEMA

Toda organización, empresa o entidad formalizada y estructurada posee una información que la define y que fundamenta todas sus operaciones diarias y esenciales. Dependiendo de la función que desempeña dicha entidad o empresa en la sociedad se define la característica de la información y el nivel de protección que se requiere para la misma.

En la Unidad de Servicios Penitenciarios y Carcelarios de Colombia, USPEC, entidad que se dedica al manejo jurídico y administrativo del sistema penitenciario y carcelario a nivel nacional, la información que se maneja no sólo es de carácter autentico y legítimo y se encuentra por ley bajo protección especial del Estado Colombiano ya que se trata de todo lo que concierne a la Población Privada de La Libertad y a la salvaguarda del Sistema Carcelario del País.

Siendo el portal WEB el medio de acceso por excelencia a la información de la Unidad, éste requiere de un sistema de acompañamiento que vele por su seguridad, integridad y disponibilidad en el acceso, manejo y manipulación de la información. No obstante, el gestor donde se aloja la información que actualmente maneja la Unidad de Servicios Penitenciarios y Carcelarios, no cumple como gestor de contenidos con los estándares de seguridad que consienta a los usuarios y al administrador acceder y descargar la información veraz y autentica de la Unidad.

Es importante mencionar que el portal web está diseñado en un sistema de gestión de contenido llamado Joomla donde todo lo que se publica queda guardado en una base datos conectado a este gestor y que en el momento no se encuentra actualizado. El portal web de la Unidad se encuentra en la versión 2.5 de Joomla y actualmente este gestor se encuentra en la versión 3.8, no ha sido posible su actualización porque tiene implementados Plugins que no dejan, ni

permiten realizar esta operación, motivo por el cual la parte administrativa se encuentra preocupada, ya que no saben qué tan seguro es su portal.

A través de los conocimientos adquiridos en este posgrado se tratará de encontrar las vulnerabilidades posibles y las consecuencias que se puedan conllevar por esta problemática, y de esta forma buscar mitigar por medio de estrategias y mecanismos de seguridad este problema.

Es necesario y fundamental para la Unidad de servicios Penitenciarios y Carcelarios del país contar con un portal WEB con estándares óptimos de seguridad, integridad y disponibilidad tanto para los usuarios finales como para los usuarios administradores en la Unidad de Servicio Penitenciarios y Carcelarios.

Si examinamos la importancia de la seguridad en el portal WEB y en especial, en el contexto que actualmente la unidad es la encargada del Sistema Carcelario Nacional del país, es un tema con un elevado grado de jerarquía, pues hoy dado los avances tecnológicos, los portales web no solo son la cara principal de la organización, sino el medio a través del cual se establecen canales de comunicación con usuarios que requieren de los servicios de una organización o establecer un contacto necesarios entre usuarios, proveedores o administradores de la misma.

Es así, como se ha estimado la necesidad a través del presente trabajo, realizar escaneos y pruebas de penetración basada en metodologías de seguridad como el proyecto OWASP en el portal web de la Unidad de Servicios Penitenciarios y Carcelarios, donde se evidencia que es de fundamental utilidad para establecer posibles fallas, malversaciones o intromisiones que afecten la seguridad del sistema.

La unidad de servicios carcelarios y penitenciarios USPEC, tiene una responsabilidad muy importante como entidad del estado puesto que debe

gestionar y suministrar bienes, prestar servicios, infraestructuras penitenciarias, asimismo debe apoyar en el ámbito logístico y administrativo que se requiere para el conveniente trabajo de los servicios penitenciarios y carcelarios a nivel nacional. Su portal web fue diseñado como instrumento tecnológico para el seguimiento, monitoreo y procesamiento de información, así como la Figura de diseños, maquetas e infraestructura de los pabellones de seguridad en Colombia.

Es así como surge la necesidad de plantearnos ¿Cuáles son los estándares que debe cumplir un portal WEB con el fin de proteger y mitigar vulnerabilidades de la Unidad de Servicios Penitenciarios y Carcelarios? ¿Cómo desde la Especialización en Seguridad Informática podemos contribuir al mejoramiento de la seguridad del portal WEB de la USPEC? ¿Cómo sebe implementarse pruebas de penetración y escaneos que ayuden a fortalecer la seguridad y cuál es el ámbito de su cobertura y eficacia?

**Figura1.Página web unidad de servicios carcelarios y penitenciarios USPEC**


Disponible en: https://www.uspec.gov.co/

# 3 JUSTIFICACIÓN

Este trabajo ha surgido como una propuesta para el mejoramiento de la seguridad del portal WEB de la Unidad de Servicios Penitenciarios y Carcelarios USPEC. Por un lado, este trabajo no sólo será un gran aporte a la mencionada entidad estatal en lo que concierne al trabajo tecnológico que busca proteger y colaborar con la seguridad de la información, sino que será de gran validez para el contexto académico ya que se pretende exponer, explicar y demostrar como a través de las buenas prácticas y conocimientos adquiriditos en seguridad informática, se puede contribuir al perfeccionamiento para mejorar aquellas fisuras de seguridad en su plataforma y evitar que usuarios no registrados y/o no autorizados, tengan acceso a información confidencial, causando alteración del portal

El contenido alojado en este portal WEB. Es tanto como para el personal interno de la institución, como para el público, una fuente de información que expone las ilustraciones, monitoreo y publicación de información relevante de la entidad. Es por esta razón adicional que se considera que deben tomarse y aplicarse medidas de contingencia tecnológica y de buenas prácticas que permitan tener un portal confiable.

Es importante señalar, la plataforma web de la unidad de servicios penitenciarios y carcelarios USPEC, se encuentra desarrollada, mediante un sistema de información CMS gestor de contenido, versión de joomla 2.5.6, pues al no estar en su última versión se encuentra expuesta a ataques lo cual permite a intrusos remotos realizar ataques de inyección de objetos PHP y ejecutar código PHP arbitrario a través del encabezado HTTP User-Agent. Fallos de restricción de acceso a URLs, suplantación de identidades de usuarios cambios no autorizados en la administración de contenidos, esto indica un intento de ataque para explotar una vulnerabilidad de PrivilegeEscalation en el componente JoomlaUser. Esta vulnerabilidad se debe a que la aplicación no ha desinfectado correctamente la

entrada de los usuarios antes de usarla en la creación de una cuenta. Como resultado, un atacante remoto puede enviar una consulta creada para crear un usuario con un permiso elevado entre otras amenazas.

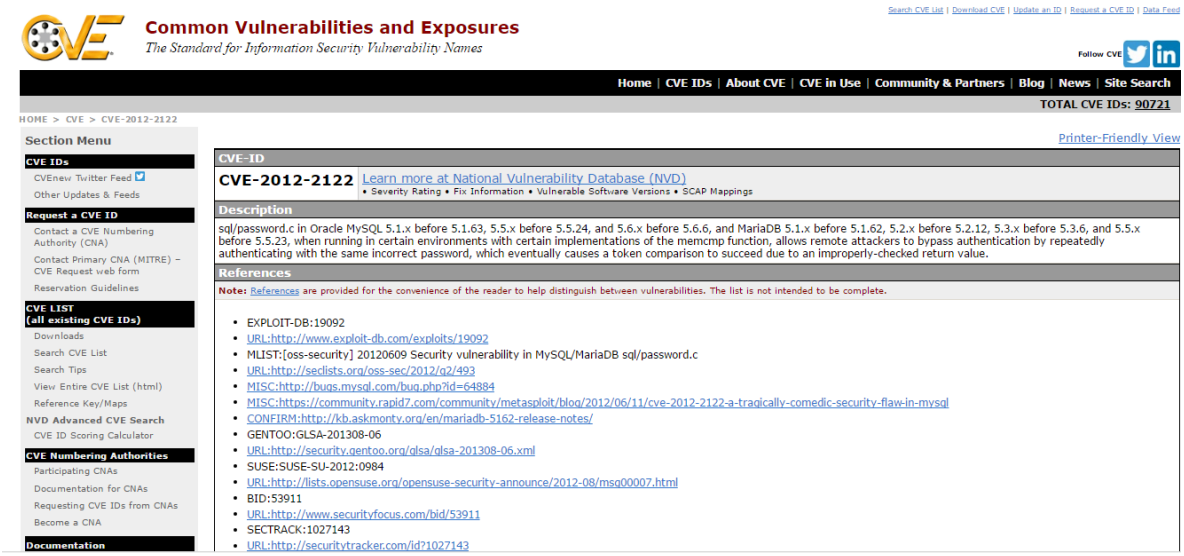
En este tipo de escenarios en plataformas web, donde la administración de contenidos es tan cambiante y constante, justifica el hecho y la necesidad de tener en cuenta unos métodos de guía que contengan planes de contingencia y seguridad basadas en las herramientas de vulnerabilidades del top 10 que OWASP ofrece.

Con esto, el grupo de investigación pretende poner en funcionamiento portal web en un gestor de contenidos que permita al portal WEB de la Unidad de Servicios Penitenciarios y Carcelarios aumentar su nivel de seguridad y establecer una información de carácter público evitando que esta sea malversada, vulnerada o manipulada sin autorización de la entidad, para fines ilegales o ilícitos.

Es en este punto, donde el trabajo encuentra su más importante justificación, ya que se define claramente la necesidad que hoy tiene la Unidad de Servicios Penitenciarios y Carcelarios en objetivos de seguridad de la información y donde se evidencia la utilidad y pertinencia de todo el objetivo del grupo con el presente trabajo. Así, podemos afirmar con contundencia que esta investigación y aplicación del proyecto es necesaria porque el portal WEB de la entidad estatal en mención, no cuenta con plena seguridad de la información ni con la evento de detectar posibles; al mismo tiempo es útil porque servirá a la Unidad de Servicios Penitenciarios y Carcelarios para proteger y salvaguardar toda la información del estado que se expone en su portal y aumentará los niveles de seguridad de la información que se tienen previstos en los objetivos de toda entidad del estado. Por último, es también pertinente ya que el objetivo principal del presente trabajo encaja de manera coherente con el problema que actualmente padece la Unidad y que se describe detalladamente en este documento en la Formulación del

Problema, por lo tanto, se pretende dar una solución eficaz con el desarrollo de este proyecto que contribuya en primera medida a mitigar la problemática de seguridad y en segunda instancia a promover la protección de la información del estado a cargo de  la Unidad de Servicios Penitenciarios y Carcelarios USPEC.

**Figura 2. Incidente de ataques reportados a la página de la USPEC.**




Fuente: Unidad de servicios carcelario y penitenciarios. Página Web.

Disponible en: http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2012-2122

# 4 OBJETIVOS

## 4.1 OBJETIVO GENERAL

Implementar medidas de seguridad mediante escaneos y pruebas de penetración basado en el proyecto OWASP para el portal WEB de la Unidad de Servicios Penitenciarios y Carcelarios USPEC con el fin de mitigar riesgos a la seguridad de la información del mismo.

## 4.2 OBJETIVOS ESPECÍFICOS

- Identificar las debilidades de seguridad del portal web de la unidad de servicios carcelarios y penitenciarios USPEC, mediante técnicas de penetración para fortalecer soporte logístico y administrativos requeridos para la apropiada labor de los servicios penitenciarios. Aplicando el proyecto de seguridad de OWASP.

- Realizar el levantamiento de la información, estado del portal, normas, mostrando procesos de identificación, rastreo y diagnóstico de los problemas de seguridad del portal web.

- Documentar y realizar las pruebas de penetración para el portal web de la unidad de servicios penitenciarios la USPEC.

- Implementar una solución para mitigar las vulnerabilidades diagnosticadas a través del proyecto OWASP en el actual portal web de la USPEC “Unidad de Servicios penitenciarios y carcelarios del país”

# 5 MARCO REFERENCIAL

## 5.1 ANTECEDENTES

El elemento más importante y central que visualiza una entidad u organización es su portal web ubicado como activo a proteger por el área de TI donde se representa la información de la unidad de servicios penitenciarios y carcelarios USPEC y se basa en la administración de contenidos y por ser prestadora de servicios se requiere una eficiente seguridad de la información.

El proyecto OWASP, Se basa en seguridad de aplicaciones web liderado por expertos en seguridad informática. El cual proporciona información imparcial, experiencia y fructífera sobre seguridad de aplicaciones informática. Relacionando lo anterior la pérdida de barreras a través del uso de facilidades de acceso remoto, exposiciones de seguridad tales como virus, intrusiones, accesos no autorizados, ataques de fuerza bruta, suplantaciones nos lleva a la necesidad de una administración efectiva de la seguridad de la información.

Los elementos asociados que se ejecutan en la red son mecanismos rápidos que proveen una alta nivel de posibilidades de comunicación, interacción y entretenimiento, que pueden ser accedidos por personas ajenas a dicha información como son los portales web. Sin embargo estos elementos deben contener mecanismos que protejan y reduzcan los riesgos de seguridad alojados, distribuidos y potencializados a través del mismo como nos indica el proyecto OWASP siendo estas políticas de seguridad, restricciones de puertos, medidas de contingencia que garanticen disminuir el riesgo de presentar eventos que se lleguen a materializar y generen impacto en el portal web.

En este sentido se busca que la seguridad del portal web USPEC mediante pruebas de vulnerabilidad increpe su sistema de contenido que se encuentra en

versiones inferiores no cumpliendo con los pilares de la seguridad informática Integridad, confidencialidad y disponibilidad que en conjunto con el cumplimiento de guías OWASP, OWASP TOP 10 y métricas de seguridad en aplicaciones fortalecen la seguridad del portal web.

La página de la USPEC está en versiones obsoletas de software, donde se identificaron diferentes aplicaciones instaladas con versiones obsoletas que ya no cuentan con soporte del fabricante y por consiguiente deben instalarse actualizaciones para mitigar vulnerabilidades. Previamente identificadas Joomla: En las direcciones 190.60.111.101 y 192.168.70.233 se tiene una versión obsoleta de Joomla, estas cuentan con CVE y exploit público. Lo anterior permite tomar control de servidor u obtener una terminal dentro del equipo.
Enlace listado de vulnerabilidades: https://drive.google.com/open?id=11jAINvFBAfxD06aSxRvTsa1RQmAoakmG

## 5.2 MARCO CONTEXTUAL

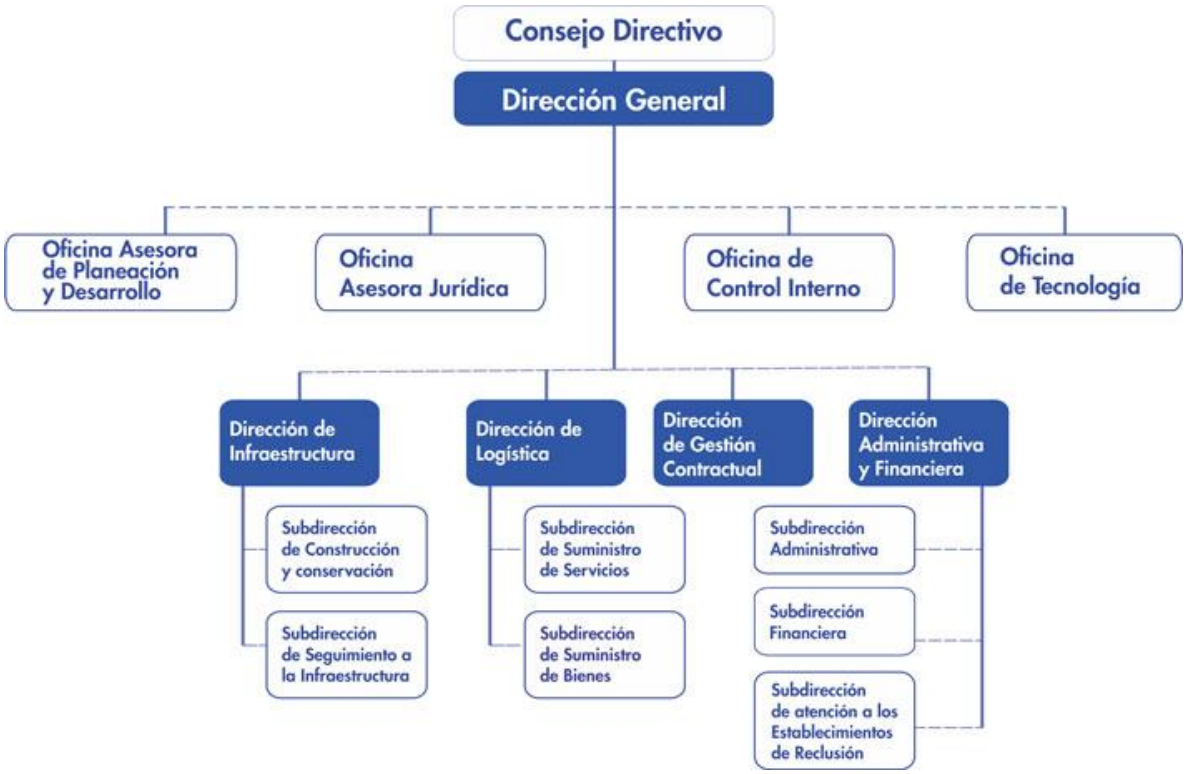
Para garantizar el respeto a la dignidad humana; el ejercicio de los derechos fundamentales y el bienestar de la población privada de la libertad en los establecimientos de reclusión, el Gobierno Nacional, a través del decreto 4150 del 3 de noviembre de 2011, creó la Unidad de Servicios Penitenciarios y Carcelarios, una entidad especializada que en consonancia con las funciones del Instituto Nacional Penitenciario y Carcelario INPEC, se concentra en la gestión y operación para el suministro de bienes y la prestación de servicios para esta población.

Tal como lo señala el artículo 4 del mismo decreto, la Unidad de Servicios Penitenciarios y Carcelarios – USPEC, tiene como objeto gestionar y operar el suministro de bienes y la prestación de los servicios y la infraestructura, y brindar el apoyo logístico y administrativo requeridos para el adecuado funcionamiento de

los servicios penitenciarios y carcelarios a cargo del Instituto Nacional Penitenciario y Carcelario – INPEC.

Posteriormente la Ley 1709 del 20 de enero de 2014, en su artículo 7 modificó el artículo 15 de la Ley 65 de 1993 determinando que el Sistema Nacional Penitenciario y Carcelario está integrado por el Ministerio de Justicia y del Derecho; el Instituto Nacional Penitenciario y Carcelario (INPEC) y la Unidad de Servicios Penitenciarios y Carcelarios (USPEC), como, adscritos al Ministerio de Justicia. Con la puesta en funcionamiento de esta nueva ley la Unidad se denomina Unidad de Servicios Penitenciarios y Carcelarios (USPEC).

**Figura 3. Organigrama USPEC**




Fuente: https://www.uspec.gov.co/?page_id=256

**Funciones de la Oficina de Tecnología**

1. Proponer al Director políticas, para optimizar el uso y aprovechamiento de la tecnología, la información y las comunicaciones intra e interinstitucionales.

2. Proponer, implementar y evaluar el plan estratégico de tecnología de la Información y las comunicaciones para la administración y gestión de la Unidad, en concordancia con las políticas y directrices del Ministerio de Tecnologías de la Información y las Comunicaciones, los lineamientos del Ministerio de Justicia y del Derecho.

3. Diseñar las políticas y programas de articulación tecnológica interinstitucional de conformidad a las directrices del Ministerio de Tecnologías de la Información y las Comunicaciones.

4. Vigilar que en los procesos tecnológicos de la entidad se tengan en cuenta los estándares y lineamientos dictados por el Ministerio de las Tecnologías de la Información y las Comunicaciones que permitan la aplicación de las políticas que en materia de información expida el Departamento Nacional de Planeación y el Departamento Administrativo Nacional de Estadísticas (DANE).

5. Formular estrategias encaminadas a la actualización y mejoramiento continuo de los recursos tecnológicos necesarios para la gestión de los establecimientos de reclusión.

6. Promover el desarrollo tecnológico mediante el impulso de mejores prácticas, técnicas, insumos y sistemas que propendan por el mejoramiento continuo de los

servicios penitenciarios y carcelarios, en coordinación con las dependencias internas y el Instituto Nacional Penitenciario y Carcelario (INPEC).

7. Desarrollar la tecnología, las redes, los sistemas de información, de seguridad y comunicacionales, que apoyen la prestación de los servicios penitenciarios y carcelarios.

8. Prestar asesoría y asistencia técnica a las dependencias y al Instituto Nacional Penitenciario y Carcelario (INPEC), en el uso y empleo de las diferentes herramientas de software, seguridad y de comunicaciones con los que cuente la Entidad.

9. Propender por la tecnificación de los sistemas de seguridad y control de la infraestructura de la gestión penitenciaria y carcelaria, en coordinación con la Dirección de Infraestructura y el Instituto Nacional Penitenciario y Carcelario (INPEC).

10. Proponer estrategias que faciliten el desarrollo de medios de seguridad, monitoreo, registro y comunicación para la mejor prestación de los servicios penitenciarios y carcelarios.

11. Elaborar estudios para la selección, adquisición e implementación de bienes y servicios en materia de tecnología que permitan apoyar y potencializar el desarrollo y la ejecución de los procesos Institucionales.

12. Implementar las acciones necesarias para asegurar la organización, administración, seguridad y control del hardware, software y de la información institucional.

13. Formular, implementar y socializar políticas de seguridad informática.

14. Investigar, promover y divulgar el uso de nuevas tecnologías para mejorar la gestión penitenciaria y carcelaria.

15. Apoyar la implementación y sostenibilidad del Sistema de Gestión Institucional y sus componentes.
16. Atender las peticiones y consultas relacionadas con asuntos de su competencia.
17. Las demás funciones asignadas que correspondan a la naturaleza de la dependencia.

## 5.3 MARCO TEÓRICO

En el proceso de desarrollo de este proyecto de investigación, está basado en cada una de las definiciones de nociones que se pretenden implementar en el campo aplicativo que son construcciones de vital importancia y fundamental para la comprensión del lector.

### 5.3.1 Portal Web.

Es el medio de acceso la forma de acceder a internet codificados en leguajes de programación y son interpretados o visualizados en navegadores. El canal bajo estándar, interpretando las solicitudes del usuario y la recepción de las páginas que proceden del servidor, igualmente, interpreta los desarrollos con lenguaje HTML y sus resultados en una página HTML combinadas con otros códigos fuente dentro de su estructura como JavaScript, Flash, PHP que conforman el entorno grafico que navegamos en internet.

### 5.3.2 Joomla Versión 2.5.6.

Joomla es un administrador de contenidos el cual permite desarrollar sitios web dinámicos e interactivos. Esta versión de Joomla 2.5.6 es una versión obsoleta, ya que Joomla actualmente está en la versión 3.8, pero como todas las versiones

esta también admite crear, modificar todas las instrucciones además de eliminar contenido de un sitio web de modo sencillo a través de un “sección de administración".

### 5.3.3 Seguridad informática.

Como bien sabemos es un área de la informática y/o computación la cual se encarga de conservar salvar y proteger la información, aplicando combinación de herramientas tanto de software, como hardware, redes, bases de datos, infraestructura, políticas o normatividad para mitigar vulnerabilidades, ya que el fruto más codiciado para cualquier organización es su información y es punto rojo para los atacantes cibernéticos.

El fortín fundamental de seguridad es garantizar Confidencialidad, Integridad y Disponibilidad: Mecanismo que garantiza fiabilidad y acceso a personal autorizado mediante mecanismos determinados por el área de tecnología. La confidencialidad es asegurar los datos, no permitiendo accesos a agentes externos de administración de gestores de contenidos y que no tienen permiso para ello. Así pues, para controlar la confidencialidad de los datos se requiere dispositivos de verificación y autorización.

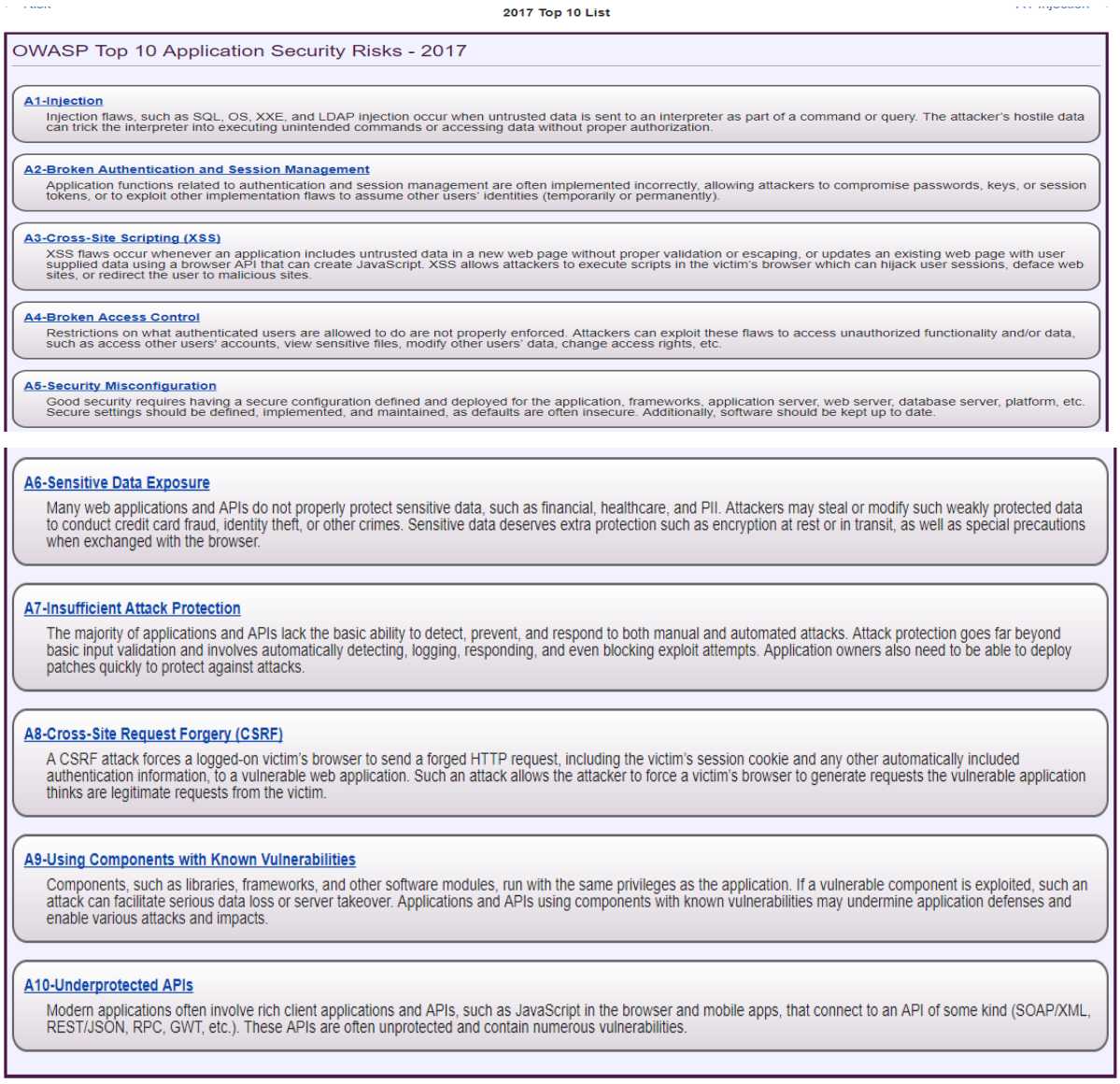
**OWASP**. El proyecto OWASP (Open Web Application Security Project) mecanismo que ayuda en el proceso del ciclo de desarrollo del software (SDLC), ésta provee una solución flexible que mejora el proceso de desarrollo, teniendo en cuenta desde el inicio el tema de la seguridad en la ingeniería del software. Procesos de identificación, rastreo y diagnóstico.

El top 10 de OWASP contempla los riesgos y vulnerabilidades en aplicaciones web más relevantes, se basa en información sobre riegos provenientes de 8 firmas especializadas en seguridad de aplicaciones. Es estos momentos se acumulan en

muchos registros de vulnerabilidades documentadas. Asimismo, estas vulnerabilidades son priorizadas de acuerdo con el nivel de explotación, detección e impacto estimado.

De esta forma se describe, el top 10, proyecto de seguridad en aplicaciones web:

**Figura 4. Top 10 Riesgos de OWASP**

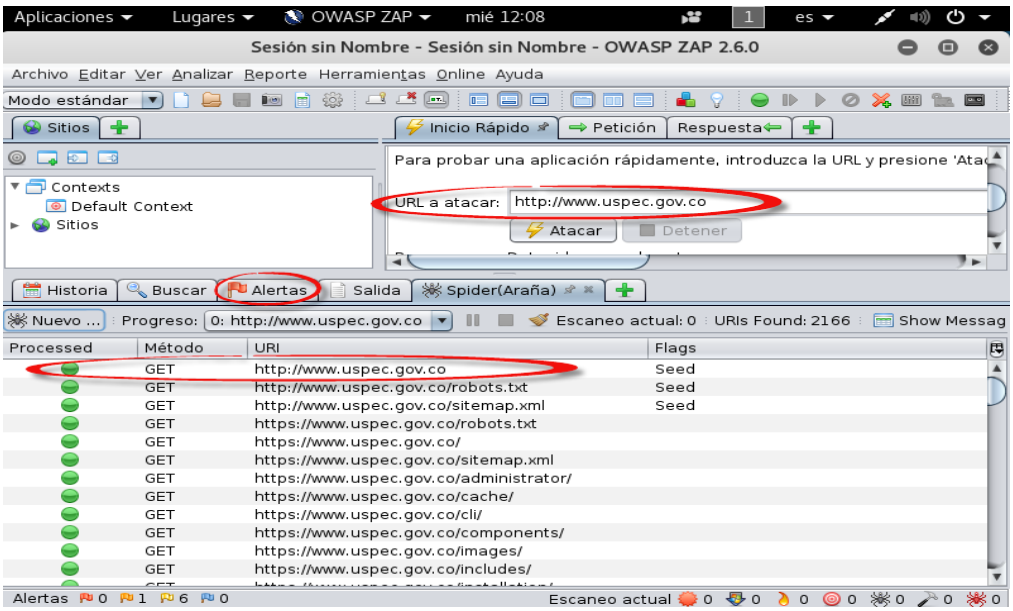



Fuente: OWASP top 10-2013

**Procesos 1. De identificación, rastreo y diagnóstico.**

Para poder comprender un poco más específicamente estas guías, como grupo de investigación se pone a prueba el instrumento contenido en sistema Kali Linux **OWASP-ZAP** (**ZedAttack Proxy**), mediante este se puede ejecutar y encontrar pruebas de vulnerabilidad, permitiendo encontrar **debilidades** en aplicaciones web, en análisis mecanizados.

**Figura 5.Uso de la herramienta OWASP-ZAP**




**Fuente: Autor**

### 5.3.4 Vulnerabilidades en la web.

Aunque en el TOP 10 de OWASP mostramos las vulnerabilidades más importantes hoy en día, vamos a explicar los conceptos en qué consisten algunas de las vulnerabilidades más comunes en general que puede sufrir las aplicaciones web y que afectan su seguridad:

**Ataques de Cross−Site Scripting**: conocido popularmente como XSS Son fallas de XSS ocurren en aplicaciones que roban información entregada por un usuario y la envía a un navegador Web a excepción de validar o codificar el contenido. XSS consiente en que atacantes ejecutan cadenas de comandos en el navegador Web de la víctima que logran retener sesiones de usuario, modificar sitios Web.

Las páginas son vulnerables a XSS cuando los datos suministrados al servidor (un comentario, un cambio en un perfil, una búsqueda, etc.) se ve posteriormente expuesto en la página de respuesta, una de las formas más sencillas de ejecutar este tipo de ataques es copiar el código entre dos etiquetas de HTML, dentro de un comentario HTML, dentro de un código de Javascript entre sus etiquetas.

Estos ataques han sido explotados para crear poderosos ataques de phishingy de abusos en el navegador donde los atacantes típicamente se valen de código HTML y de scripts ejecutados en el cliente.

“A partir de la liberación del lenguaje JavaScript, se previeron los riesgos de admitir a un servidor Web enviar código ejecutable al navegador. Un problema se presenta cuando los usuarios tienen abiertos varias ventanas de navegador, en algunos casos un script de una página podría acceder datos en otra página, observando el riesgo de que un sitio malicioso pretendiera acceder datos sensibles de esta forma.

*Política same−origin*: Fundamentalmente permite la interacción entre objetos y páginas, mientras estos objetos vengan del mismo dominio y en el mismo protocolo. Impidiendo así que un sitio malicioso posea acceso a datos sensibles en otra ventana del navegador vía JavaScript. Se introducen otros componentes y políticas de control en los navegadores y en los lenguajes en el lado del cliente,

para proteger a los usuarios de sitios maliciosos.  XSS alcanzan ser distinguidas a modo técnicas de evasión de las políticas de protección.

Las inconstantes XSS suelen ser para mostrar alguna información tipo IP, User-Agent y demás. Por eso las webs que brindan servicios agregados como el de mostrar el nombre de usuario autenticado, mostrar la IP, zonas horarias, datos de fecha de inicio de sesión entre otros, datos que suelen ser vulnerables cuando se alteran las cabeceras[1].

En resumen, el atacante explota vulnerabilidades de una aplicación web inyectando scripts (tipo JavaScript o VBScript por lo general) en el código fuente de los aplicativos en el lado del cliente, destruyendo, modificando y robando información de un portal web. Tipos principales deXSS:

- No perseverante o reflejado
- perseverante o almacenado.

El **no *perseverante* o *reflejado***. Es muy común la raíz de la vulnerabilidad, es el manejo inapropiado (falta de validación) de solicitudes de datos HTTP por el código del servidor, permitiendo a los sitios maliciosos reflejar código malicioso y atacar a otros usuarios. El principal vector de ataque es usualmente un mensaje de correo que contiene una URL maliciosa, cuando el usuario da clic en la URL, el código malicioso es ejecutado. Esta vulnerabilidad aprovecha el concepto de arquitectura cliente servidor (Servidor WEB Navegador Web), el navegador ejecuta el código porque cree que es el código original y no uno alterado.

[1]AMAYA TARAZONA, Carlos Alberto. ”Unidad 1 Seguridad en Aplicaciones Web “.internet: (http://datateca.unad.edu.co/contenidos/233008/AVA_2014_2/MATERIAL/UNIDAD1/REFBIBLIOREQ/UNIDAD1_SEGURIDAD_EN_APLICACIONES_WEB_2014.pdf)

El ***perseverante o almacenado***. Este ataque no requiere clic en una URL con el fin de ejecutar código malicioso. En este caso el código es capaz de vivir en el servidor vulnerable y está embebido en el código HTML. Una vez más, este tipo de ataque es el resultado directo de validaciones pobres en el lado del servidor, lo que permite forzar entradas maliciosas que pueden ser mostradas en el sitio web. Este tipo de ataque es particularmente riesgoso, no solo porque no requiere una intervención directa del usuario sino porque tiene un alcance global más peligroso[2].

- **INYECCIONES SQL:**

Se interpreta como el mecanismo o técnica que utiliza un atacante para lograr vulnerar un sitio web o los servidores mediante la inyección de códigos cuando los datos que un usuario registra son interpretados como parte de una orden o consulta. Los atacantes obstaculizan para que ejecute comandos no intencionados facilitando datos principalmente alterados.

Técnica para explotar aplicaciones web cuando las medidas de seguridad por parte de los ingenieros administradores del sistema dan cabida a que se hagan procesos de inyección que no admiten la información entregada por el cliente, para crear consultas SQL maliciosas. Ejemplo se inyecta el comando (para solicitar información).

***SELECT id FROM usuarios WHERE user='$f_user'***
***AND password='$f_pass';***

[2]ASCENCIO MENDOZA, Martha. MORENO PATIÑO, Pedro Julián. Desarrollo de una Propuesta Metodológica para Determinar la Seguridad en una Aplicación Web". Disponible en http://repositorio.utp.edu.co/dspace/bitstream/11059/2511/1/0058A811.pdf.

Si no se identifica un usuario (campo en blanco en la validación) o si se cambian las medidas predeterminados de los usuarios: *admin, administrator, guest, invitado* entre otros, al modificar la inyección de SQL por:

*$f_user= "' or 1=1 --" en las que se usa:*
*; para ejecutar múltiples queries*
*-- para comentar el final del query*
*construcciones del tipo ´ or ´´=´*
*construcciones del tipo numeroor 1=1*

Se podría explotar esta vulnerabilidad de validación de usuario e ingresar al sistema[3].

- **DENEGACIÓN DE SERVICIO (DoS):**

Vulnerabilidad usada por atacantes para lograr consumir todos los recursos informáticos de un sistema, por medio solicitudes simultáneas, terminando recursos como CPU, Memoria, acceso a la red, que imposibilitan el acceso al sistema. Este ataque tiene una variante llamada Ataque de Denegación de Servicio Distribuido (DDoS), permitiendo así atacar varios computadores o servidores y deteniendo los sistemas en ejecución[4]

- **CSRF:**

Se perpetúa forzando al navegador web en uso o atacar, validado en algún servicio una petición a una aplicación web vulnerable.

---

[3]AMAYA TARAZONA, Carlos Alberto. Op. Cit.

[4]ASCENCIO MENDOZA, Martha. MORENO PATIÑO, Pedro Julián.Op. Cit.

Esta aplicación se encarga de realizar la acción elegida a través de la víctima, debido que la actividad maliciosa será procesada en nombre del usuario autenticado. Al contrario de los ataques conocidos como Cross Site Scripting (su traducción sería ordenes en sitios cruzados – XSS) los cuales explotan la confianza del usuario para con un sitio particular; el Cross SiteRequestForgery explota la confianza que un sitio web tiene en un usuario particular.

La técnica de CSRF. Consiste en realizar acciones no deseadas sobre un dominio desde otro. Un sitio web que permitía transacciones económicas utiliza URL engañosos para la compra de objetos con la tarjeta de crédito almacenada:

Si un usuario malintencionado lograse que un usuario autenticado en la página anterior realizase clic sobre el enlace, obtendría como resultado la compra del objeto en cuestión. Esto es, podría engañar a la gente para comprar objetos que él pusiera a la venta por un precio desorbitado.

Se mejora la seguridad frente a un posible ataque podemos hacer uso de las cabeceras Referer. Estas indican la página desde la que se ha llegado a otra. Se utilizan, por ejemplo, para conocer cuáles son las búsquedas que permiten a un usuario llegar a un sitio web desde un buscador. Una medida de protección, aunque se puede saltar, sería comprobar que las peticiones realizadas a nuestras páginas, o al menos a aquellas que impliquen acciones sensibles, se realicen desde nuestro propio dominio.

Los sistemas web pueden protegerse de ataques CSRF estableciendo una serie de valores numéricos que se generen de manera única en cada petición. Estos pueden ser determinados como un CAPTCHA hace que se introduzcan instrucciones que deben digitar se para realizar acciones críticas, Estas medidas, casi siempre nos permiten asegurar los datos de usuarios.

Generalmente las aplicaciones son vulnerables si no cuentan con estándares requeridos basado en el proyecto OWASP, los CSRF, toman una serie de precauciones que intenten evitar ataques mediante esta técnica. Las medidas son:

• Cerrar la sesión inmediatamente tras el uso de una aplicación.
• No permitir que el navegador almacene las credenciales de ninguna página, ni que ningún servidor mantenga nuestra sesión recordada más que durante el tiempo de uso.
• Utilizar navegadores distintos para las aplicaciones de ocio y las críticas.

Se debe realizar limpieza de las *cookies* de sesión entre navegadores para garantizar fiabilidad en los navegadores[5].

- **CR/LF INJECTIONS:**

Es una vulnerabilidad **(HTTP Splitting)** en la cual las cabeceras con caracteres especiales trabajan en unidos a los CR (CarriageRetun) y LF (Line Feed) que fundamentalmente realizan un “Salto de línea” en el instante de una conversación o salto de cabecera. En programación las señales de salto de línea se crean con los caracteres de escape \r\n.

Las aplicaciones CR/LF injection son vulnerables cuando no filtra de forma educada las variables, accediendo escanear los valores prohibidos como lo son éstos. Esta vulnerabilidad implica el que un usuario perverso pueda manejar a su perspectiva las cabeceras de un servidor.

---

[5]ALONSO CEBRIÁN, José María. GUZMÁN SACRISTÁN, Antonio. LAGUNA DURÁN, Pedro. MARTÍN BAILÓN, Alejandro. Ataques a aplicaciones web. Disponible en: https://www.exabyteinformatica.com/uoc/Informatica/Seguridad_en_bases_de_datos/Seguridad_en_bases_de_datos_(Modulo_2).pdf. .

Relacionando lo anterior al identificar los saltos de línea final, provocará la "partición" de la cabecera, permitiendo colocar junto al doble salto de línea el código, normalmente será una segunda cabecera.

- **INCLUSIÓN REMOTA DE ARCHIVOS:**

Esta vulnerabilidad se ejecuta mediante código remoto dentro de la aplicación vulnerable. Cargando un fichero de forma local, para su inclusión dentro de la página, el cual puede ser malicioso. Es un fallo que deja que se logre ejecutar cualquier código no deseado en el servidor web con los riesgos de suspender servicios. Coexisten ficheros dentro del flujo de ejecución de la aplicación web. Van desde un simple intérprete de comandos a una completa shell equipada con su propio explorador de ficheros, opciones para subir o descargar ficheros e incluso la posibilidad de ejecutar programas en el equipo remoto[6].

- **EJECUCIÓN DE COMANDOS (Command Execution):**

Este tipo de vulnerabilidad toma ventaja de la falta de validación en las entradas en un sitio web, donde el atacante puede correr comandos del sistema operativo en la aplicación web vulnerada. Generalmente, esta vulnerabilidad permite aprovechar, que los datos de usuario son pasados como parámetros a operaciones de entrada y salida, para así añadir comandos de sistema operativo por medio de caracteres especiales como pipe[7].

---

[6]ALONSO CEBRIÁN, José María. GUZMÁN SACRISTÁN, Antonio. LAGUNA DURÁN, Pedro. MARTÍN BAILÓN, Alejandro. Op. Cit.

[7]ASCENCIO MENDOZA, Martha. MORENO PATIÑO, Pedro Julián.Op. Cit.

- **SNIFEO Y CODIFICACIÓN DE CABECERAS:**

En las codificaciones de cabeceras se requiere de una serie de utilidades cuya finalidad es la de interceptar las cabeceras que envían al navegador, para acceder a su análisis o su modificación. Estas utilidades se denominan sniffers, y funcionan interceptando el tráfico que pasa por el puerto (HTTP).

Entre las herramientas se destacan las que se pueden agregar como Addons o Plugins para diferentes navegadores, que entre los cuales destacan para FireFox, Tamper Data y Live HTTP Headers, o también pueden ser programas independientes propietarios o de código abierto. Estas herramientas son fundamentales para el trabajo con las cabeceras http.

Hay diferentes formas de sniffear las cabeceras para modificarlas, con software que hagan transacciones TCP, Telnet, Putty, o los populares Netcat y CryptCat[8].

- **HTTP FINGERPRINTING:**

Este ataque consiste en conseguir información de un sistema concreto y de alguna vulnerabilidad específica. Esto es posible si se identifica el rastro del TCP/IP de los equipos atacados. Fingerprinting técnica que brinda información dentro de las muchas opciones que tiene de descubrir datos de la víctima es la de permitir descubrir de forma muy fiable el sistema operativo que se ejecuta en la maquina analizada. Consiste en la ejecución de 4 tests (orden de campos al hacer un HEAD, respuesta ante un DELETE, respuesta ante una versión del protocolo HTTP incorrecta, y por último, la respuesta que da el servidor ante un protocolo erróneo).

[8]AMAYA TARAZONA, Carlos Alberto. Op. Cit.

Con los resultados de los cuatro test, se puede diferenciar perfectamente entre servidores Apache y los IIS, ya que de cada uno se extrae un resultado diferente ante cada uno de los test. Además de las versiones de sistema operativo, puertos abiertos y capas de seguridad implementadas al protocolo HTTP. La información que puede brindar esta técnica dentro de las muchas opciones que tiene de descubrir datos de la víctima son:

- Permitir descubrir de forma muy fiable el sistema operativo que se ejecuta en la maquina analizada.
- Identificar el tipo de servidor y la versión en la que se soporta el servicio
- El tipo de servicio que se ejecuta y la versión.

De las Técnicas que existen para levantar este tipo de información con respecto al levantamiento de una huella identificativa, es el uso de "*Escuchas de Red"* o escáner de puertos como la herramienta *nmap*, con miras a establecer un posible ataque.

Relacionando lo anterior utilizando herramientas como "**hping2**, se pueden realizar ataques para: evaluar el desempeño de la red utilizando diferentes protocolos, tamaños de paquetes, TOS (typeofservice, o sea, tipo de servicio), y fragmentación; realizar descubrimiento de camino utilizando el campo MTU (onda traceroute); transferir archivos (incluso ante reglas de firewall muy fascistas); realizar funciones al estilo 'traceroute' pero bajo diferentes protocolos; detección remota de OS ('remote OS fingerprinting'); auditar una implementación de TCP/IP (`TCP/IP stack') en particular; etc. hping2 es una buena herramienta para aprender acerca de TCP/IP[9].

[9]AMAYA TARAZONA, Carlos Alberto. Op. Cit.

- **WEBTROJANS:**

Son funcionalidades desarrolladas en páginas web medianamente complejas es la posibilidad de subir ficheros al servidor: imágenes, documentos en PDF, ficheros de vídeo, etc.

Cuando envían ficheros se debe comprobar que lo que se recibe un fichero legítimo, es decir, si estamos esperando la llegada de un fichero con una imagen, es necesario comprobar que lo que se recibe es realmente una imagen y no otro tipo de fichero[10].

- **INYECCIONES DE PHP:**

Son ataques que fructifican el evento de la aplicación de escalar archivos al servidor. Estos ataques funcionan de la siguiente manera:

Corrientemente PHP almacena los archivos subidos en una carpeta temporal, sin embargo, es común en las aplicaciones cambiar la localización del archivo subido a una carpeta permanente y leerlo en la memoria. Al hacer este tipo de instrucciones se debe validar el parámetro que hará referencia al nombre del archivo, ya que puede ser truqueado a modo de apuntar a archivos de configuración del sistema (como /etc/passwd en sistemas Unix).

El momento de utilizar comando que puedan ser interpretados en aplicaciones web se interpreta según la forma de invocarlos, es posible que un usuario

[10]ALONSO CEBRIÁN, José María. GUZMÁN SACRISTÁN, Antonio. LAGUNA DURÁN, Pedro. MARTÍN BAILÓN, Alejandro. Op. Cit

malicioso logre ejecutar un comando externo distinto al esperado. Ej.: uso del carácter ";" en Unix o "&" en Windows[11].

### 5.3.5 Norma ISO 27001

Es una norma internacional emitida por la Organización Internacional de Normalización (ISO) y describe cómo gestionar la seguridad de la información en una empresa.

Se enfoca principalmente en proteger la confidencialidad, integridad y disponibilidad de la información en una empresa. Esto lo hace, mediante la evaluación de riesgos potenciales que pueden alterarse afectando la información, y luego definiendo controles y procedimientos para mitigar o tratar el riesgo. Por lo tanto, la filosofía principal de la norma ISO 27001 se basa en la gestión de riesgos: investigar dónde están los riesgos y luego tratarlos sistemáticamente.

La revisión más reciente de esta norma es la ISO/IEC 27001:2013 surge con base en la revisión realizada del estándar ISO 27001 y publicada en ese mismo año. Fue la primera revisión que se realizó a la norma ISO 27001:2005, debido a que es una estructura de alto nivel, según las tendencias de los sistemas integrados de gestión, y a la alineación y articulación con la norma ISO 31000 de Gestión de Riesgos. Entre las principales ventajas de esta versión, tenemos:

- Flexibilidad para implementación en todo tipo de empresas.
- Facilidad de integración con otras normas de gestión relacionadas.

### 5.3.6 Norma ISO 27002

Las políticas y normas de seguridad se hacen eficientes si tienen Control los directores quienes las revisan y las hacer poner en práctica en las organizaciones con regularidad para el cumplimiento del procesamiento y procedimientos de

[11] AMAYA TARAZONA, Carlos Alberto. Op. Cit.

información dentro de su área de responsabilidad, con las políticas y normas de seguridad apropiadas, y cualquier otro requisito de seguridad.

Básicamente la ISO 27002 sugiere que las organizaciones cumplan con los controles que establecen de seguridad informática para mitigar los riesgos que se puedan presentar en aplicaciones web y tener un sistema robusto que perita protegerse de muchas vulnerabilidades.

Identificar las causas de la no conformidad, y evaluar la necesidad de acciones para lograr cumplimiento.

Implementar las acciones correctivas apropiadas.

Revisar la acción correctiva tomada, para verificar su eficacia e identificar cualquier fortaleza o debilidad.

## 5.4 MARCO CONCEPTUAL

### 5.4.1 Información

Conjunto de datos controlados, sistemáticos y procesados que completan un mensaje sobre unos determinados activos. Se considera un activo de gran importancia por lo que requiere mayor protección.

### 5.4.2 Seguridad de la Información

Es el interés en que se garantice la confidencialidad, integridad y disponibilidad de la información; además, otras propiedades como autenticidad, responsabilidad, no repudio y fiabilidad pueden ser también consideradas para evitar su perdida.[12]

### 5.4.3 Los Activos

Son todos los elementos con los que se interactúan las empresas: Los datos o información, servicios, aplicaciones (software), equipos (hardware), recursos físicos y recursos humanos.

### 5.4.4 Amenazas

Se determina a la escena que se puede presentar como usuarios, programas maliciosos, errores de programación, programas maliciosa y catástrofes naturales que dañen los activos de información, mediante la explotación de una vulnerabilidad.

### 5.4.5 Vulnerabilidad

Son las debilidades que tiene una empresa, lo cual hace que se presenten riesgos o amenazas que a través de ellas afecten los servicios del sistema.

### 5.4.6 Impacto

Se puede ver en el suceso que se producen en un activo cuando ocurre una amenaza.

---

[12]SISTEMADE GESTIÓNDESEGURIDADDELAINFORMACIÓN, Tomado el día 15 de noviembredel2017,[En línea].http://www.iso27000.es/download/doc_sgsi_all.pdf

### 5.4.7 Los Controles

Son las medidas de protección que se implementan en una empresa para minimizar los riesgos informáticos que se puedan generar.

### 5.4.8 Políticas de Seguridad

Maneras medidas, controles y estrategias que permiten proteger los activos de información de una empresa.

### 5.4.9 Incidente de seguridad

Serie de eventos de seguridad no deseados o inesperados que tienen una significativa probabilidad de comprometer las operaciones activas y/o información dela empresa.

### 5.4.10 Gestión de activos

Busca proteger los activos de información, controlando el acceso solo a las personas que tienen permiso de acceder a los mismos.

### 5.4.11 Seguridad física

Consiste primariamente en prevenir el acceso no autorizado a las instalaciones para prevenir daños o pérdidas de activos o hurto de información.

### 5.4.12 Sistema de Gestión de Seguridad de la Información (SGSI)

Sistema de preservación de su confidencialidad, integridad y disponibilidad, así como de los métodos implicados en su tratamiento, dentro de una organización.

### 5.4.13 Riesgo

Estimación del grado de exposición a que una amenaza se materialice sobre uno o más activos causando daños o perjuicios a la Organización, el riesgo indica lo que le podría pasar a los activos si no se protegen adecuadamente.[13]

### 5.4.14 Análisis del Riesgo

Proceso sistemático para estimar la magnitud de los riesgos a que está expuesta una organización.

### 5.4.15 Riesgo Potencial

Es el daño probable de un activo cuando se encuentra desprotegido.

### 5.4.16 Pentest

Es un ataque a un sistema informático con la intención de encontrar las debilidades de seguridad y todo lo que podría tener acceso a ella, su funcionalidad y datos.

### 5.4.17 Sitio web

Espacio virtual formados por una colección de páginas web relacionadas y son accesibles a un dominio de internet.

---

[13]SISTEMADE GESTIÓNDESEGURIDADDELAINFORMACIÓN, Tomado el día 15 de noviembredel2017, [En línea].http://www.iso27000.es/download/doc_sgsi_all.pdf

### 5.4.18 Aplicación web

Herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de internet o de una intranet mediante un navegador y esto a su vez permite procesar los datos o archivos almacenados dentro de la web.

## 5.5 MARCO LEGAL

Basados en las normas legales de la república de Colombia y documentación de seguridad del proyecto OWASP enfocamos este documento para promover la calidad.

**Ley 1273 del 5 enero 2009** Capitulo primero de los atentados contra la confidencialidad, la integridad y la disponibilidad de los datos y de los sistemas informáticos

**Artículo 269a**: acceso abusivo a un sistema informático. El que, sin autorización o por fuera de lo acordado, acceda en todo o en parte a un sistema informático protegido o no con una medida de seguridad, o se mantenga dentro del mismo en contra de la voluntad de quien tenga el legítimo derecho a excluirlo, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes. [14]

**Artículo 269b:** obstaculización ilegítima de sistema informático o red de telecomunicación. El que, sin estar facultado para ello, impida u obstaculice el funcionamiento o el acceso normal a un sistema informático, a los datos

[14] Ley1273 de 05 enero 2009. {en línea} {11 de noviembre de 2017} Disponible en: http://www.mintic.gov.co/portal/604/articles-3705_documento.pdf

informáticos allí contenidos, o a una red de telecomunicaciones, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes, siempre que la conducta no constituya delito sancionado con una pena mayor.

**Artículo 269c:** interceptación de datos informáticos. El que, sin orden judicial previa intercepte datos informáticos en su origen, destino o en el interior de un sistema informático, o las emisiones electromagnéticas provenientes de un sistema informático que los trasporte incurrirá en pena de prisión de treinta y seis (36) a setenta y dos (72) meses.

**Artículo 269d:** daño informático. El que, sin estar facultado para ello, destruya, dañe, borre, deteriore, altere o suprima datos informáticos, o un sistema de tratamiento de información o sus partes o componentes lógicos, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes.

**Artículo 269e**: uso de software malicioso. El que, sin estar facultado para ello, produzca, trafique, adquiera, distribuya, venda, envíe, introduzca o extraiga del territorio nacional software malicioso u otros programas de computación de efectos da Flinos, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes.

**Artículo 269f:** violación de datos personales. El que, sin estar facultado para ello, con provecho propio o de un tercero, obtenga, compile, sustraiga, ofrezca, venda, intercambie, envíe, compre, intercepte, divulgue, modifique p emplee códigos personales, datos personales contenidos en ficheros, archivos, bases de datos o medios semejantes, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa

y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes.

**Artículo 269g:** suplantación de sitios web para capturar datos personales. El que con objeto ilícito y sin estar facultado para ello, diseñe, desarrolle, trafique, venda, ejecute, programe o envíe páginas electrónicas, enlaces o ventanas emergentes, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes, siempre que la conducta no constituya delito sancionado con pena más grave. En la misma sanción incurrirá el que modifique el sistema de resolución de nombres de dominio, de tal manera que haga entrar al usuario a una IP diferente en la creencia de que acceda a su banco o a otro sitio personal o de confianza, siempre que la conducta no constituya delito sancionado con pena más grave. La pena señalada en los dos incisos anteriores se agravará de una tercera parte a la mitad, si para consumarlo el agente ha reclutado víctimas en la cadena del delito.

**Artículo 269h:** circunstancias de agravación punitiva: las penas imponibles de acuerdo con los artículos descritos en este título se aumentarán de la mitad a las tres cuartas partes si la conducta se cometiere: 1. Sobre redes o sistemas informáticos o de comunicaciones estatales u oficiales o del sector financiero, nacionales o extranjeros. 2. Por servidor público en ejercicio de sus funciones 3. Aprovechando la confianza depositada por el poseedor de la información o por quien tuviere un vínculo contractual con este. 4. Revelando o dando a conocer el contenido de la información en perjuicio de otro. 5. Obteniendo provecho para sí o para un tercero. 6. Con fines terroristas o generando riesgo para la seguridad o defensa nacional. 7. Utilizando como instrumento a un tercero de buena fe. 8. Si quien incurre en estas conductas es el responsable de la administración, manejo o control de dicha información, además se le impondrá hasta por tres años, la pena de inhabilitación para el ejercicio de profesión relacionada con sistemas de información procesada con equipos computacionales.

**Artículo 2691:** hurto por medios informáticos y semejantes. El que, superando medidas de seguridad informáticas, realice la conducta señalada en el artículo 239 manipulando un sistema informático, una red de sistema electrónico, telemático u otro medio semejante, o suplantando a un usuario ante los sistemas de autenticación y de autorización establecidos, incurrirá en las penas señaladas en el artículo 240 de este Código.

**Artículo 269j:** transferencia no consentida de activos. El que, con ánimo de lucro y valiéndose de alguna manipulación informática o artificio semejante, consiga la transferencia no consentida de cualquier activo en perjuicio de un tercero, siempre que la conducta no constituya delito sancionado con pena más grave, incurrirá en pena de prisión de cuarenta y ocho (48) a ciento veinte (120) meses y en multa de 200 a 1500 salarios mínimos legales mensuales vigentes. La misma sanción se le impondrá a quien fabrique, introduzca, posea o facilite programa de computador destinado a la comisión del delito descrito en el inciso anterior, o de una estafa. Si la conducta descrita en los dos incisos anteriores tuviere una cuantía superior a 200 salarios mínimos legales mensuales, la sanción allí señalada se incrementará en la mitad.

## 5.6 MARCO TECNOLÓGICO

Mostraremos un breve resumen de las herramientas software que podemos usar para realizar el análisis de la seguridad web del portal basado en el proyecto OWASP:

### 5.6.1 Distribución de sistema operativo para penttesting o pruebas de intrusión.

Kali Linux es una comercialización especializada en el campo de la seguridad informática y posee herramientas de pruebas de intrusión o penttesting, basada en el Sistemas Operativos Linux, y ayuda a ejecutar las valoraciones necesarias a sistemas de información, para encontrar que problemas de seguridad que nos afecta. Está basada en Debian, y fue diseñada especialmente para la auditoria y seguridad informática en general. En este momento es mantenida por Offensive Security Ltd. que desarrolló la distribución a partir de la reescritura de BackTrack

### 5.6.2 Herramientas para analizar y explotar vulnerabilidades web.

#### 5.6.2.1 Nmap

Aplicación gratuita y de código abierto, asume un papel importante que permite descripciones de red, administración de programaciones de actualización de servicio y monitoreo de tiempo de actividad de host o servicio. Es de las principales herramientas para ayudar la seguridad en sistemas informáticos.

#### 5.6.2.2 Nessus

Es una herramienta escaneo de vulnerabilidades, se identifica por tener alta rapidez de hallazgo, auditoria en la configuración de aplicaciones, revelando datos sensibles para análisis de vulnerabilidades de la red.

#### 5.6.2.3 Wireshark

Importante por su capacidad en realizar un escaneo profundo de la red, a nivel de los paquetes que viajan por ella, Wireshark permite analizador de paquetes y protocolos, gratuito y de código abierto, nos permite analizar y auditar con el máximo detalle nuestra red local, pudiendo tanto encontrar vulnerabilidades como posibles problemas de red.

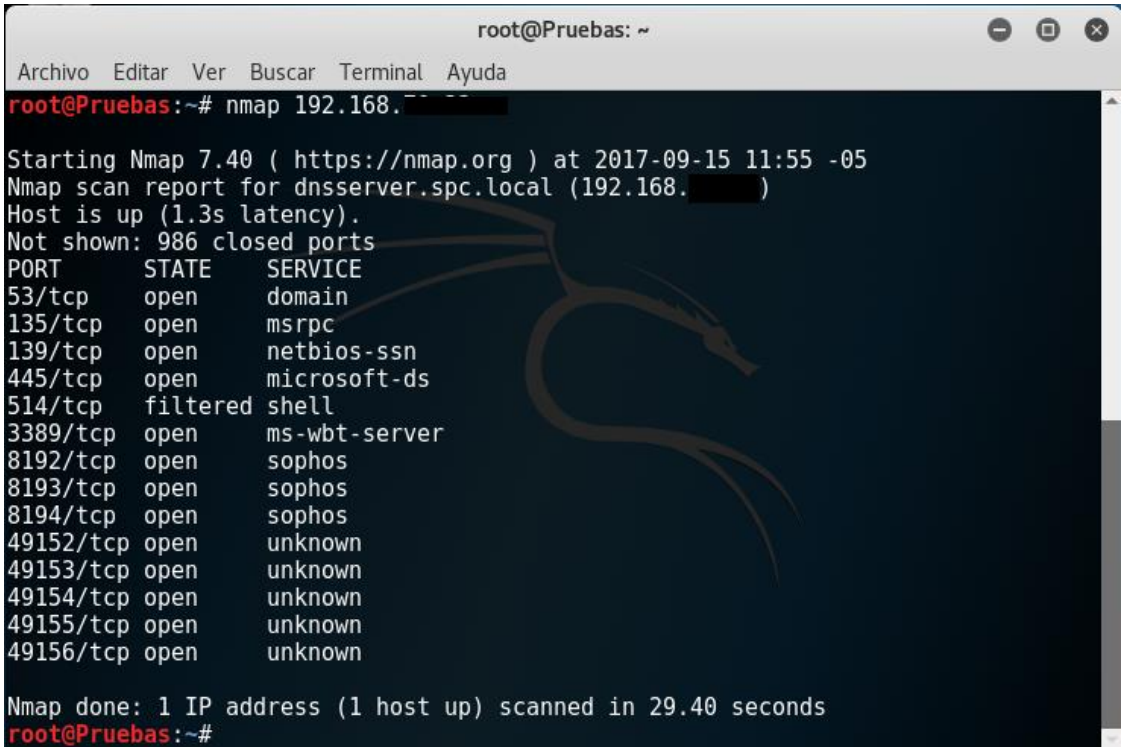
#### 5.6.2.4 Metasploit PenetrationTesting Software

La mayoría de los exploits que vienen con Metasploit para explotar vulnerabilidades conocidas con los que se puede aprovechar de distintas vulnerabilidades por sus potentes simulaciones de ataques tanto en sistemas informáticos como en redes. Gracias a ella, se puede comprobar si el sistema se encuentra totalmente actualizado y correctamente protegido frente a estas vulnerabilidades que, de ser descubiertas, pueden llegar a exponer seguridad en los sistemas.

#### 5.6.2.5 OWASP Zed

Herramienta de Linux que permite realizar múltiples pruebas y a su vez encontrar vulnerabilidades en aplicaciones web, en lugar de hacerlo sobre redes o sobre sistemas físicos. Como son sus análisis activos o pasivos sobre los enlaces y sitios visitados, validando

**Figura6. Ejecución de NMAP**




Fuente: Autor

#### 5.6.2.6 Xsser

Es un framework que automatiza la detección y explotación de vulnerabilidades XSS inyección de código no deseado en aplicaciones web. Con esto un atacante podrá cambiar la actuación de la aplicación, con lo que se logran corromper datos.

#### 5.6.2.7 Fimap

Funciona auditando y explotando errores de inclusión local y remota de archivos en aplicaciones web.

#### 5.6.2.8 Csrftester

Herramienta de código abierto desarrollada en JAVA que actúa como un servidor proxy que limpia las solicitudes HTTP en el navegador Web en busca de vulnerabilidades de tipo CSRF, para ello toca instalar la máquina virtual de JAVA.

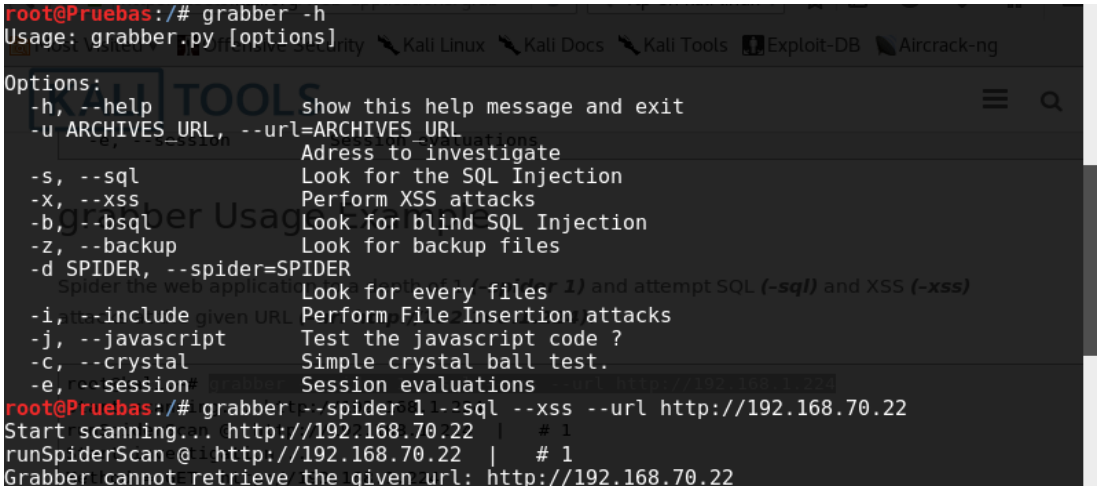
#### 5.6.2.9 OWASP Mantra

Es unas herramientas libres y de código abierto, integradas en un navegador, la totalidad de estas herramientas son populares como complementos o extensiones del navegador, viene dentro de la Suite de Backtrack o Kali linux.

#### 5.6.2.10 Grabber

Es un escáner web. Fundamentalmente descubre las vulnerabilidades estableciendo el sitio web que se requiere auditar.

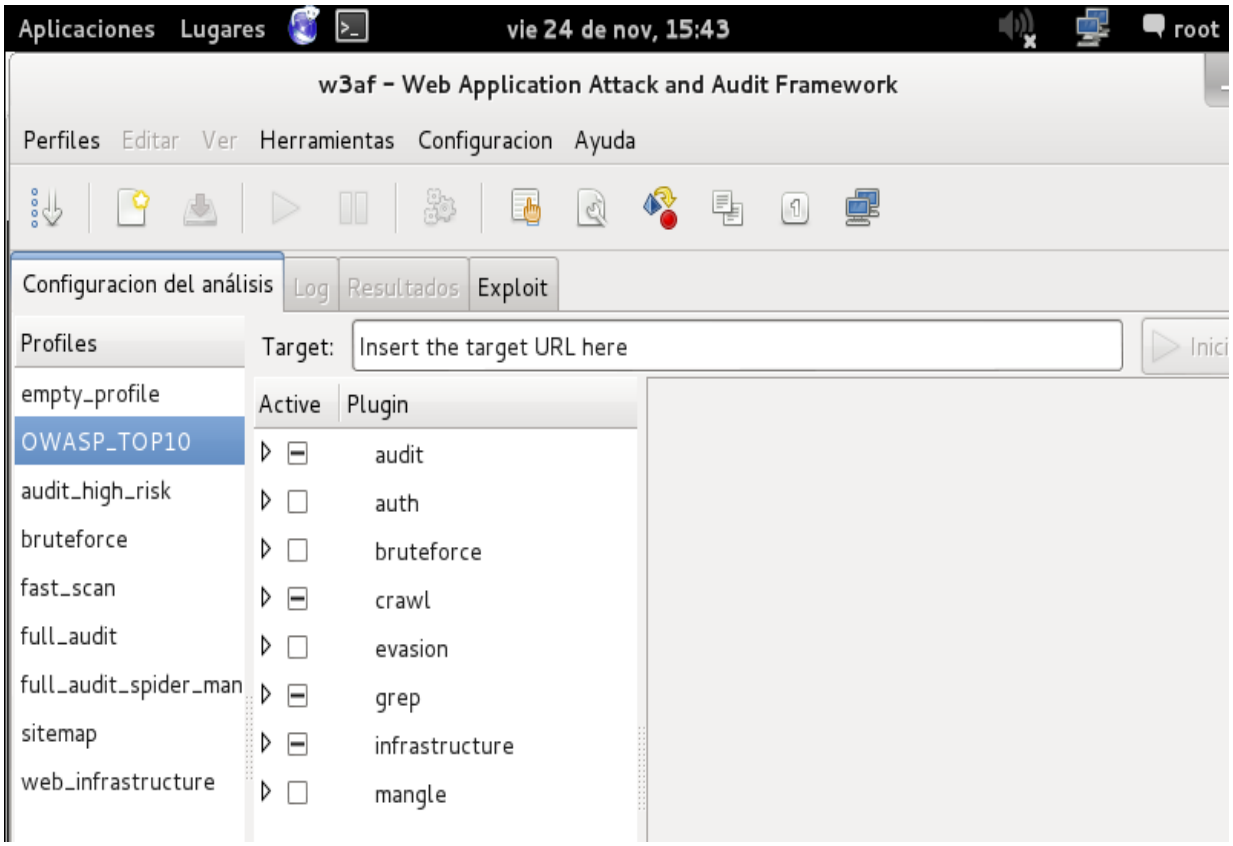
**Figura7. Aplicación Grabber detectando una falla de seguridad.**




Fuente: Autor

**W3AF:** Es un marco de auditoría para aplicaciones web mediante la búsqueda y explotación de todas las vulnerabilidades que puede contener una aplicación web, identifica más de 200 vulnerabilidades en aplicaciones web incluso la más conocidas como inyección de SQL, XSS, errores de configuración entre otras, es desarrollado en el lenguaje programación Python, es open source, está en la suite de Kali Linux y también sirve de ayuda el proyecto de seguridad de OWASP.

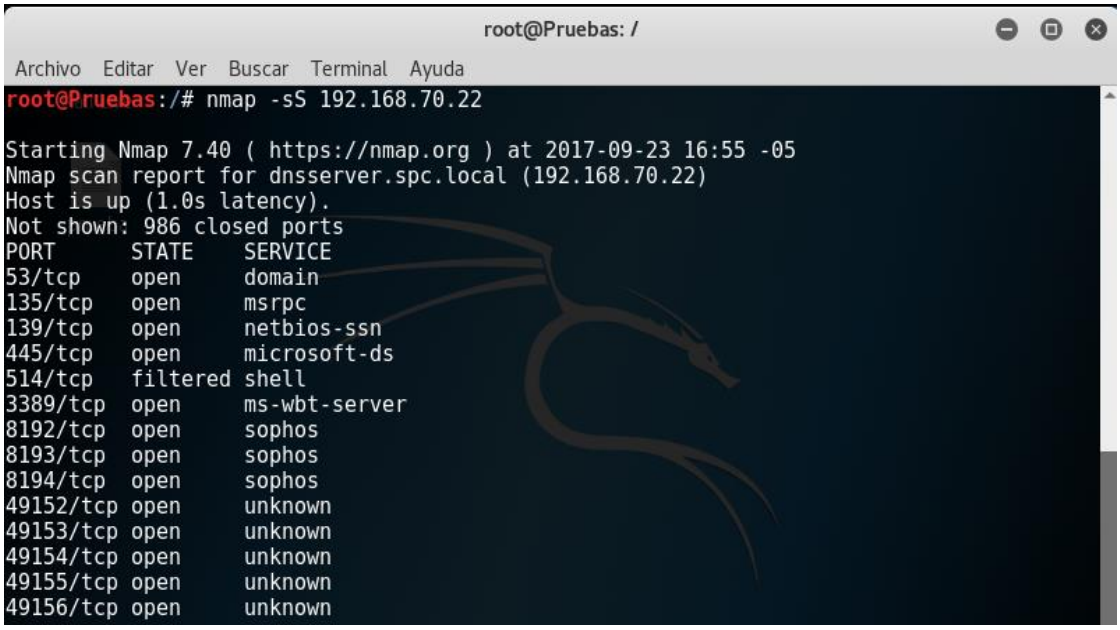
**Figura 8. Herramienta W3AF y sus perfiles de escaneo**




Fuente: Autor

**NMAP**: Nos sirve para escanear que puertos están abiertos así como el tipo de sistema operativo que tiene un host y sus servicios es una herramienta muy útil para labores de auditoría en seguridad web y de penttesting.

**Figura 9.Nmap detectando puertos abiertos.**




Fuente: Autor

# 6 DISEÑO METODOLÓGICO

La investigación que se pretende desarrollar para el proyecto pentesting para el portal web de la uspec, apoyado en el proyecto de seguridad OWASP. Encaja dentro de lo que se define como una investigación exploratoria ya que realizaremos investigación aplicada al consultar directamente los problemas de seguridad encontrados el portal web USPEC. De manera que se procura hacer el control de las vulnerabilidades, amenazas y riesgos es por ello que debemos garantizar la confidencialidad, integridad y disponibilidad de la información del portal.

Contando con apoyo de la dependencia de tecnología para indagar los aspectos más relevantes del el portal a investigar y buscarle solución a sus problemas de seguridad, también decimos que es una investigación mixta porque aparte de hablar con los implicados en los problemas del portal web también tomaremos datos de otras fuentes documentales y autores sobre el proyecto OWASP para aplicar los componentes de este proyecto en el análisis y solución de problemas de seguridad en el portal.

Según el problema a resolver podemos decir que nuestra investigación será aplicada a resolver los problemas de seguridad de la (plataforma web) USPEC, aplicando principios teóricos del proyecto OWASP que es un proyecto abierto de seguridad en aplicaciones Web que contiene métodos o guías que son: Guía para la construcción de aplicaciones Web y Servicios Web Seguros, Guía de pruebas, herramienta para la educación y concienciación, en materia de seguridad, en las aplicaciones web. Según la intención se puede definir como una investigación por que averigua un problema el cual se ha presentado refiriéndose a las vulnerabilidades web del portal USPEC y mirar cómo solucionarlas.

## 6.1 FUENTES, TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE INFORMACIÓN.

Las técnicas de investigación están definidas como un conjunto de procedimientos y herramientas para recoger, generar analizar y presentar información, de modo que para el presente trabajo se utilizaran las siguientes fuentes:

**Tipos de fuente:** Las fuentes que vamos a tomar son las primarias porque consultaremos directamente a los administradores del portal y ellos son los que nos van a dar información de primera mano del caso en particular.

**Técnicas**: Utilizaremos instrumentos cuantitativos como entrevistas, cuestionarios, listas de chequeo para preguntarle a los implicados de la administración del portal aspectos técnicos y de funcionamiento del portal y averiguar sus principales fallas de seguridad que podría tener según el testimonio dado por los funcionarios y personas que acceden al portal. Como instrumentos cualitativos podemos hacer un estudio del caso llevando registros escritos o audiovisuales sobre el funcionamiento del portal, incluso podemos aplicar la observación al encargado de administrar la página para ver como lo hace y analizar aspectos de la seguridad en el portal, también analizaremos documentación concerniente al proyecto OWASP y cómo podemos aplicarlo en el análisis y solución de los problemas de seguridad del portal.

**Instrumentos:** Como punto de apoyo tomaremos información de fuente segundarias que son libros, artículos de páginas web, foros, blogs videos, audios entre otros para mostrar aspectos de OWASP en la solución del caso que es la seguridad del portal. También dentro de los instrumentos de investigación podemos usar herramientas software, estas herramientas software vienen integradas y son parte del proyecto OWASP y mostraremos cuales nos servirán para el análisis y solución de aspectos de seguridad del portal.

# 7 DESARROLLO DEL PROYECTO

## 7.1 ASPECTOS TEÓRICOS FASES DE PRUEBAS

### ¿Qué es una prueba de intrusión de aplicación web?

Procedimiento de evaluar la seguridad de un sistema de computadores o una red mediante la simulación de un ataque. Una prueba de intrusión para el portal web de la unidad de servicios carcelarios y penitenciarios está enfocada solamente a evaluar la seguridad de dicha aplicación web.

Se valida mediante un análisis activo de la aplicación en busca de cualquier debilidad, fallos técnicos o vulnerabilidades en la misma. Cualquier incidencia de seguridad que sea encontrada será presentada al administrador del sitio web, en este caso, será la dependencia de Tecnología de la Unidad de servicios carcelarios y penitenciarios, a la misma entidad se le suministrara una evaluación del impacto y una propuesta para su mitigación o una solución técnica.

### 7.1.1 ¿Qué es una vulnerabilidad?

Las aplicación web posee un conjunto de activos (recursos de valor como los datos en una base de datos o en el sistema de archivos), para el caso del portal, uno de los activos más importante resulta ser la peticiones de los reclusos o información de los funcionarios administrativos, así mismo resulta importante mencionar que una vulnerabilidad es una debilidad en un activo que hace posible a una amenaza. Así que una amenaza es un caso potencial que puede dañar un activo (**funcionamiento intranet, pqrd, divulgación de información herrada**), mediante la explotación de una vulnerabilidad. Un test es una acción que tiende a mostrar una vulnerabilidad en la aplicación web.

### 7.1.2 ¿Qué es la metodología de pruebas OWASP?

Las pruebas de intrusión no se conciben ser exactos, sin embargo, mediante algunas se puede definir una lista completa de todas las incidencias posibles que deberían ser comprobadas. Ante todo, sistema las pruebas de intrusión son solo una técnica apropiada para comprobar la seguridad de aplicaciones web bajo ciertas circunstancias.

El objetivo es recopilar todas las técnicas de comprobación posibles, explicarlas y mantener la guía actualizada. La metodología de pruebas de intrusión de aplicaciones web usando OWASP se basa en un enfoque de acercamiento de la caja negra. La persona que realiza las pruebas tiene poca, o ninguna, información sobre la aplicación que va a ser comprobada.

En el modelo de pruebas al portal web de la unidad de servicios carcelarios y de la USPEC consta de:

- **Auditores**: Personal que se encarga de realizar tareas de Pent-test y comprobaciones.
- **Herramientas y metodología**: El núcleo de este proyecto de pruebas de penetración.
- **Aplicaciones:** Pruebas de caja Negra, Blanca y Gris sobre la cual se realizarán las pruebas.

**Las pruebas se dividen en 2 fases:**

**Modo pasivo:** Al realizar las pruebas se pretende intuir en la lógica de la aplicación, puede usarse una utilidad para la recopilación de información, como un proxy HTTP, para observar todas las peticiones y respuestas HTTP. En esta fase se debería percibir cuales son todos los puntos de acceso (puertas) de la aplicación (p.e. cabeceras HTTP, parámetros, cookies).

**Modo activo:** En esta fase se realizan las pruebas usando la sistemática descrita en los siguientes apartados.
Para el caso del portal web, se divide en las pruebas subcategorías:

- Pruebas de gestión de la configuración
- Pruebas de Autenticación
- Pruebas de Autorización
- Pruebas de gestión de sesiones
- Pruebas de validación de datos
- Pruebas de denegación de Servicio
- Pruebas de Servicios Web
- Pruebas de AJAX

### 7.1.3 Evaluación de penetración de aplicaciones Web.

Comprendemos las considerables limitaciones de las herramientas de pruebas automatizadas, como escáneres de aplicaciones Web, de modo que casi todas nuestras pruebas se realizan y se verifican manualmente, usando una metodología bien definida, constante y coherente. Utilizamos herramientas automatizadas en áreas de la evaluación solamente donde hemos comprobado que son precisas y efectivas (generalmente, en menos del 5% del servicio) y hemos patrocinado un proyecto de investigación OWASP para medir el rendimiento de estas aplicaciones automatizadas.

**Descubrimiento**: trabajamos junto a usted para comprender el impacto en el negocio de diversas características, de modo que podamos calificar y cuantificar el riesgo que las vulnerabilidades detectadas presentan para el negocio.

**Evaluación**: para asegurarnos de realizar pruebas en todas las áreas fundamentales y para garantizar la consistencia, usamos un marco de seguridad común que incluye lo siguiente:

- Autenticación
- Autorización
- Administración de usuarios
- Administración de sesiones
- Validación de datos, incluidos todos los ataques comunes como inyección de SQL, scripting entre sitios, inyección de comandos y validación del lado del cliente
- Administración de manejo y excepción de errores
- Auditoría y registros

**Informes y resultados**: Al finalizar el servicio, redactamos un informe detallado con un resumen ejecutivo que da prioridad de los descubrimientos y explica el impacto para su negocio. Todos nuestros descubrimientos técnicos individuales contienen detalles específicos y recomendaciones para su mitigación[15]

### 7.1.4 Técnicas para detección de vulnerabilidades.

De acuerdo con los resultados obtenidos de la revisión sistemática sobre las herramientas y técnicas más utilizadas actualmente para detección de vulnerabilidades, se pueden establecer las siguientes:

## 7.2 DESARROLLO FASE DE PRUEBAS

[15]McAffe.Evaluación de penetración de aplicaciones Web. Disponible en: (http://www.mcafee.com/mx/services/technology-consulting/software-and-application-security-services/web-application-penetration-assessment.aspx#vt=vtab-Overview).

Se explica resumidamente en qué consisten algunas de estas pruebas y mostrar algunos ejemplos.

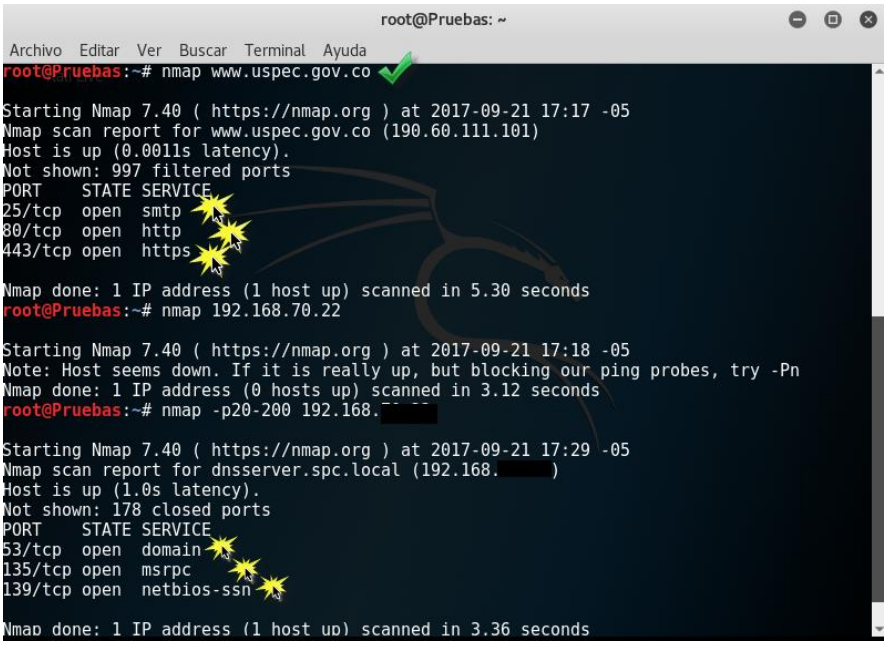
### 7.2.1 Recopilación de la información.

En fase de prueba se puede recopilar toda la información posible que pueda tener el portal web de la unidad de servicios carcelarios y penitenciarios USPEC, ya que la fase de recopilación es punto de partida para saber que posibles vulnerabilidades puede tener la página web, además evidenciar por medio de ciertas herramientas algunas características técnicas que pueda tener.

**Información de puertos abiertos:**

Una de las formas de saber que puertos abiertos tiene la página web es usar la herramienta NMAP, colocamos dentro de NMAP la orden así: nmapwww.uspec.gov.co

**Figura10. Nmap detectando puertos abiertos del portal web USPEC.**



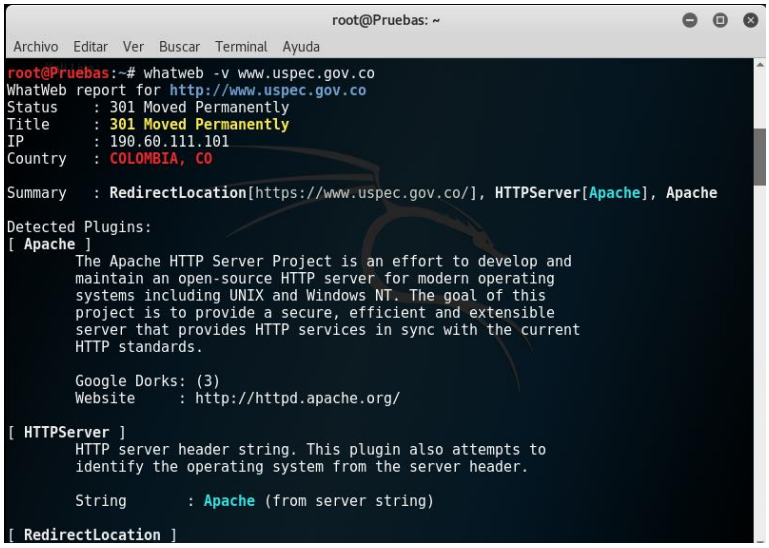

Fuente: El Autor.

Se puede visualizar en la Figura los puertos abiertos, además de la dirección IP del portal web USPEC.

**Que tecnologías web usa nuestra página:**

Una forma de saberlo es usando una herramienta de la suite de Kali Linux llamada whatweb la cual nos permite reconoce que tipos de tecnologías están contenidas en una aplicación web por ejemplo un sistema de gestión de contenidos, le tecleamos la siguiente orden para saber datos más relevantes de nuestra página: whatweb –v www.uspec.gov.co

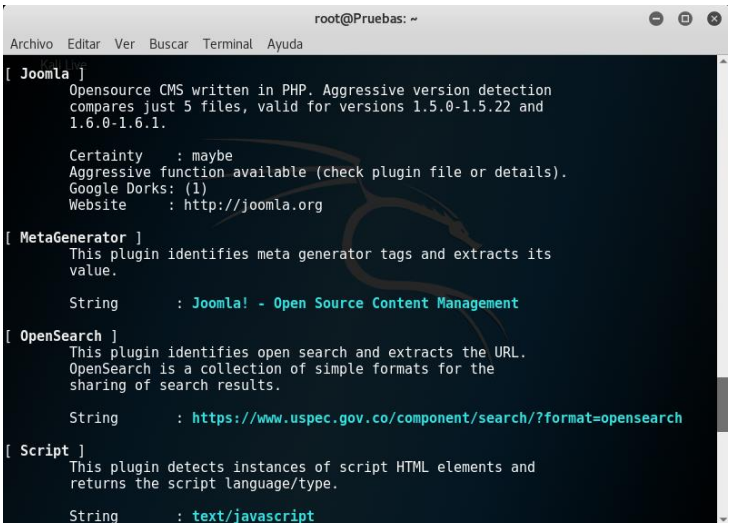
**Figura 11. WhatWeb en ejecución.**




Fuente: El Autor

Otro tipo de información importante que se puede obtener utilizando esta herramienta es tipo de servidor, cookies, dirección IP, gestor de contenido usado entre otros datos.

**Figura12. WhatWeb mostrando resultados**

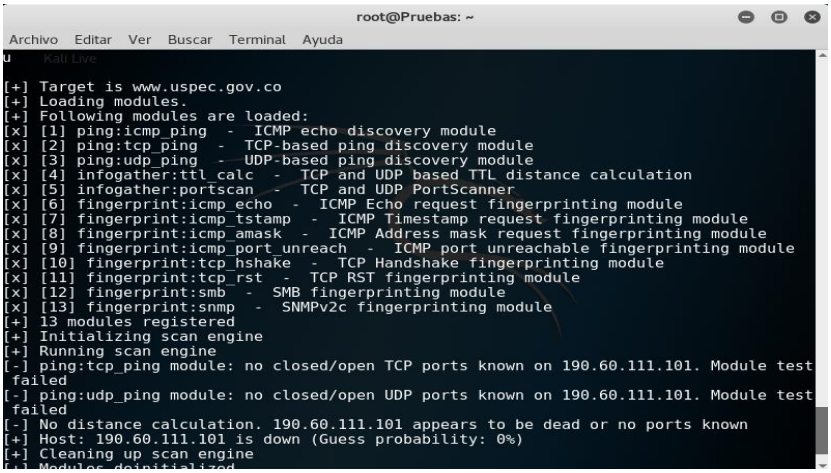



Fuente: El Autor.

- **Tipo de sistema operativo usado en el host:**

Lo podemos saber usando la herramienta xprobe2 contenida en Kali Linux, dentro del terminal de Kali Linux tecleamos la siguiente orden: xprobe2 –v www.uspec.gov.co, con el –v podemos tener información más detallada:

**Figura13. Uso del xprobe2**




Fuente: El Autor.

**Información de la página web en línea:**

Utilizando la dirección del portal web USPEC en la página web de netcraft más datos técnicos dándole clic al siguiente link: http://toolbar.netcraft.com/

http://toolbar.netcraft.com/site_report?url=www.uspec.gov.co#last_reboot Luego se debe colocar en dice lookupanotherURL, dentro de esa barra de búsqueda colocamos la dirección web del portal uspec.gov.co.

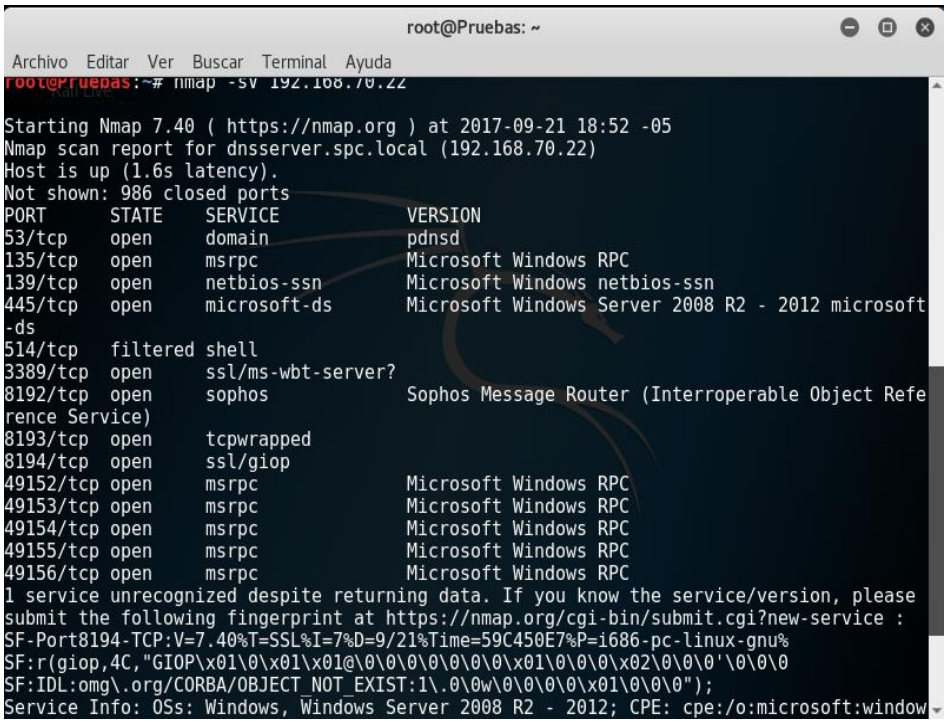
Para ver los resultados encontrados en el portal web USPEC. Clic en este link. http://toolbar.netcraft.com/site_report?url=www.uspec.gov.co#last_reboot.

**Descubrimiento de las aplicaciones**

Para esto se debe probar la existencia de las aplicaciones en los puertos:

Usando en la herramienta Kali Linux NMAP con opción –sV para identificar los servicios https asociados a esos puertos analizados: dándole la siguiente orden: nmap –sV 192.168.70.22, -sV sirve para detectar servicios.

**Figura14.servicios asociados a los puertos.**



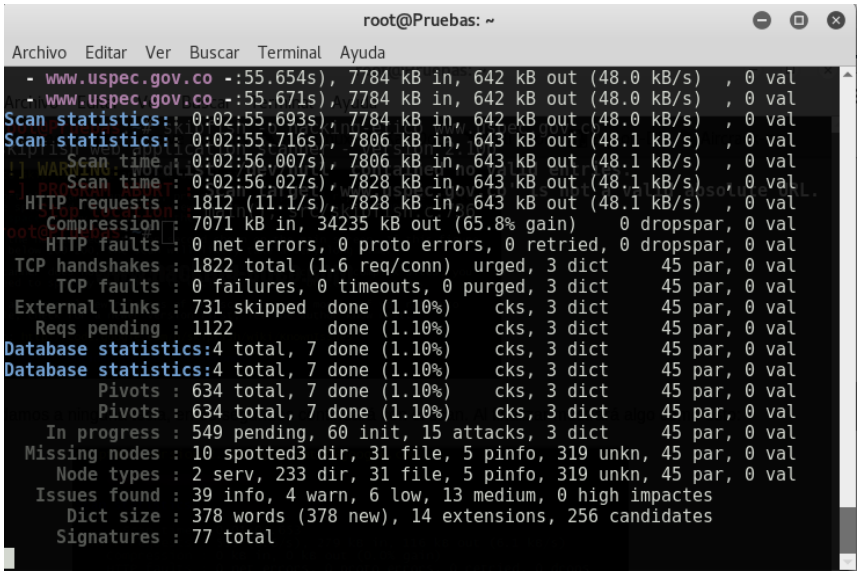

Fuente: EL Autor.

- **Skipfis:**

El oficio principal de este tipo de herramientas es realizar pruebas de conceptos sobre algún sitio Web, para ver qué resultados nos muestra, creando un resumen en HTML de los contenidos. La dirección web a la que se realiza la prueba de

auditoría con la siguiente en la instrucción: *skipfish -o hacking-ético https://www.uspec.gov.co/* “-o” es opción de reporting, seguido del nombre que queremos darle al directorio cuando finalice las pruebas y continuando con la URL Un ejemplo al usar la herramienta skipfish se pueden obtener resultados y reporte de pentest.

**Figura15. Resultados del skipfish.**

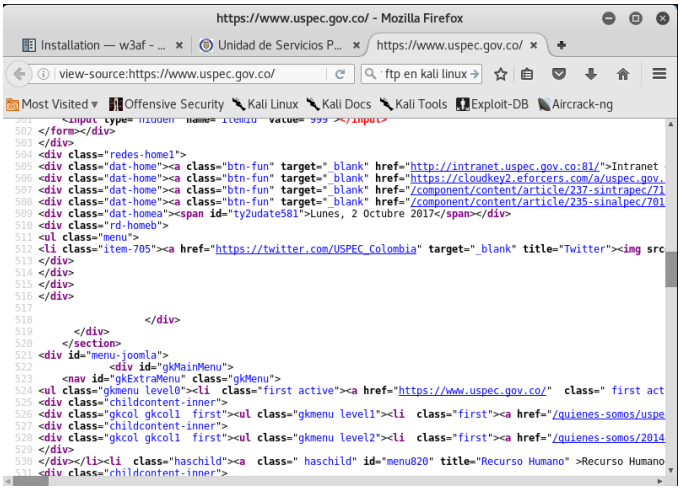



Fuente: Autor

**Identificación de puntos de entrada de la aplicación**

Una manera para ver esto es revisar el código fuente de la aplicación Por ejemplo, con la herramienta OWASP Mantra para revisar código:

**Figura16. Inspección de código portal web USPEC.**




Fuente: El Autor

En esta captura podemos observar el código fuente del portal web de la USPEC, Incluso vemos que tiene la opción para editar el código fuente, aunque por lo general para este tipo de revisiones de código fuente se necesita gran conocimiento en los lenguajes de programación ya que al cambiar líneas de código sin el debido conocimiento se puede sabotear el contenido de la página.

**Otras pruebas para tener en cuenta a futuro:**

- Pruebas de firmas de aplicaciones Web.
- Descubrimiento de aplicaciones.
- Análisis de código de error.

### 7.2.2 Pruebas de gestión de la configuración de la aplicación.

Hay aplicaciones web, algunas esconden alguna información que no se toma en consideración regularmente durante el desarrollo o configuración de la propia aplicación, Estos datos logran ser descubiertos en el código fuente, en archivos de registro o a través de los códigos de error por defecto de los servidores web. Una

aproximación adecuada a esta posibilidad es fundamental durante una evaluación de seguridad.
Un ejemplo muy usual es el de revisión de comentarios el incluir comentarios detallados en su código fuente permite otros programadores comprender mejor porque fue tomada una decisión específica a la hora de codificar una función dada. Sin embargo, los comentarios incluidos en línea en código HTML podrían revelar información interna, que no debería estar disponible, a un atacante potencial, incluso el código fuente queda convertido en comentario cuando ya no es necesario, pero este comentario es filtrado de forma no intencionada dentro de las páginas HTML retornadas a los usuarios.

La revisión de comentarios debería ser realizada para determinar si está siendo filtrada cualquier información a través de comentarios. Esta revisión solo puede ser realizada minuciosamente mediante un análisis del contenido estático y dinámico del servidor web, y a través de búsquedas de archivo.

Relacionando lo anterior se debe revisar las configuraciones de las aplicaciones la configuración recomendada varía dependiendo de la política del site y de la funcionalidad que deba ser provista por el software de servidor. En la mayoría de casos, no obstante, deberían seguirse las directrices de configuración (proporcionadas por el vendedor de software o bien por terceras partes) para determinar si el servidor ha sido protegido adecuadamente, incluso debemos revisar el registro ya  la intención principal de los registros de aplicación es producir una salida de operación que podría ser usada por el programador para analizar un error concreto una forma seria revisar los mensajes de error 40x (no encontrado), un gran número de estos mensajes desde el mismo origen podrían ser indicadores de una herramienta de scanner CGI siendo utilizada contra el servidor de una aplicación  ,también revisar los  Mensajes del tipo 50x (error de servidor),por ejemplo, las primeras fases de un ataque de inyección SQL

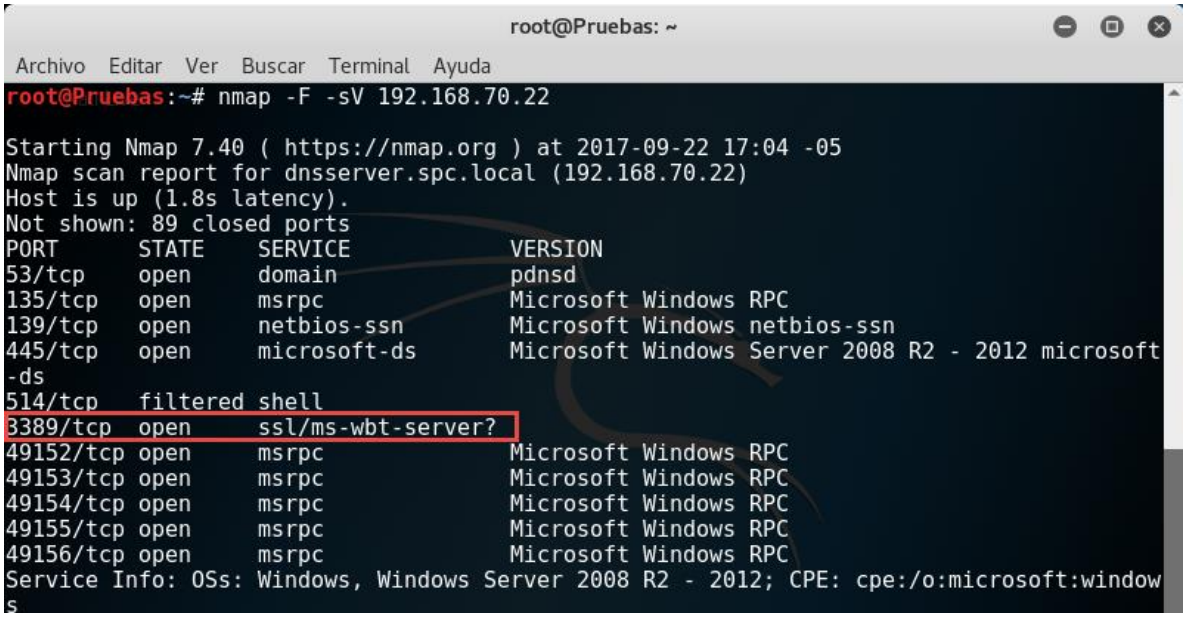
producirán este mensaje de error cuando la consulta SQL no está construida correctamente y su ejecución falla en la base de datos de backend.

**Pruebas de SSL/TLS:**

SSL y TLS son dos protocolos que hacen uso de certificados digitales para establecer seguras comunicaciones, con soporte criptográfico, canales seguros para la protección, confidencialidad y autenticación sobre la información que se transmite.

Con estas implementaciones de seguridad, es importante verificar la utilización de un algoritmo de cifrado fuerte, y su correcta implementación. Que detectar el posible soporte de cifrados débiles, se deben identificar los puertos asociados a los servicios de una aplicación, en nuestro caso será el portal web de la USPEC, una forma de saberlo es usando NMAP y la función –sV para identificar servicios asociados, nos muestra por ejemplo el protocolo SSH el cual permite realizar comunicaciones cifradas en este caso está en el puerto 3389.

**Figura17.Identificación protocolo SSH.**



Fuente: El Autor.

**Otras pruebas para tener en cuenta a futuro:**

- Pruebas del receptor de escucha de la Base de Datos.
- Pruebas de gestión de configuración de la infraestructura.
- Pruebas de gestión de configuración de la aplicación.
- Gestión de extensiones de archivos.
- Archivos antiguos, copias de seguridad y sin referencias.
- Interfaces de administración de la infraestructura y de la aplicación.
- Métodos HTTP y XST.

### 7.2.3 Pruebas de Autenticación.

Un sistema o usuario validamos en el momento que ratificamos su origen y ver si es real o si es un potencial atacante. El campo de la informática, la autenticación es el proceso de intentar comprobar la identificación digital del remitente de una comunicación. Un ejemplo común es el proceso de registro en un sistema. Al comprobar el sistema de autenticación necesitamos comprender como funciona el proceso de autenticación y usar esa información para eludir el mecanismo de autenticación de la página web a auditar.

**Pruebas para tener en cuenta a futuro:**

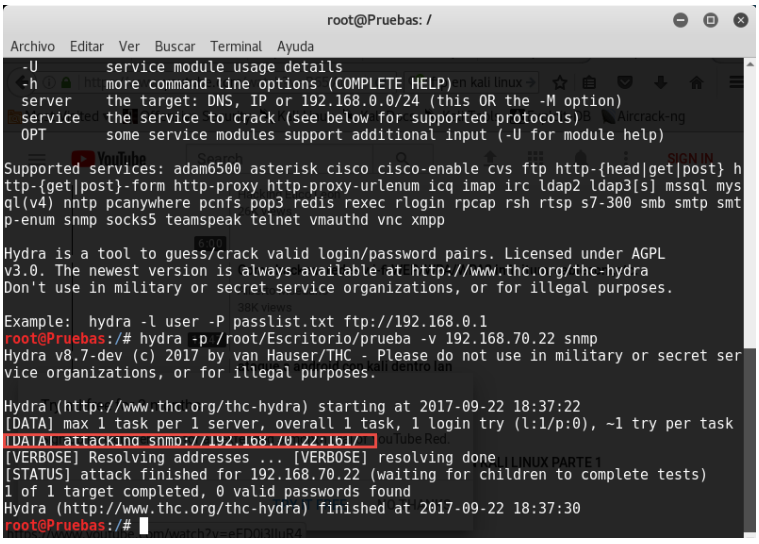
- Transmisión de credenciales a través de un canal cifrado.
- Enumeración de usuarios.
- Pruebas de fuerza bruta.
- Saltarse el sistema de Autenticación.
- Comprobar Sistemas de recordatorio/restauración de contraseñas vulnerables.
- Pruebas de gestión del Caché de Navegación y de salida de sesión.
- Pruebas de CAPTCHA.
- Múltiples factores de autenticación.
- Análisis de condiciones de carrera.

### 7.2.4 Pruebas de fuerza bruta.

Es un ataque que consiste en comprobar todas las posibilidades o combinaciones para solucionar problema en este caso el descifrar alguna contraseña, y comprobar para cada una de ellas si repara el problema expuesto. En las pruebas de aplicaciones web, la mayor parte de las veces está relacionada con la necesidad de disponer de una cuenta de usuario válida para acceder a la parte interna de la aplicación y comprobar diferentes tipos de sistema de autenticación y la efectividad de diferentes ataques de fuerza bruta.

Una herramienta que sirve para esta labor es hydra y está contenida en la suite de Kali Linux.

**Figura18.Uso de HYDRA.**




Fuente: El Autor

En esta imagen se usó hydra para atacar los servicios de SNMP de la página, pero una de las desventajas de la fuerza bruta es que este tipo de ataques toman tiempo en ejecutarse (varias horas) , además consume recursos del sistema , para que este tipo de ataques tengan más éxito se deben combinar con ataques del

tipo diccionario que este caso es un bloc de notas en el que se han almacenado una lista de palabras o frases muy usadas para contraseñas y que al ejecutarse ese diccionario nos ayude a buscar la clave.

Otros programas a tener en cuenta para este tipo de ataques son Medusa o John TheRipper.[16]

### 7.2.5 Pruebas de Autorización.

Son las que se encargan de permitir acceso a los recursos dispuestos en el sistema únicamente a los que el administrador designe que puedan acceder.

Se pretender ver comprobando si es posible eludir la autorización de la aplicación, si existe una vulnerabilidad en el traspaso de rutas o si es posible realizar un escalado de privilegios.1*

**Pruebas para tener en cuenta a futuro:**

- Pruebas de path transversal.
- Pruebas para saltarse el esquema de autenticación.
- Pruebas de escalado de privilegios.

### 7.2.6 Pruebas de gestión de sesiones.

Los controles que se realizan sobre el usuario, desde la autenticación hasta la salida de la aplicación. HTTP es un protocolo sin estados, lo que significa que los servidores web responden a las peticiones de clientes sin enlazarlas entre sí, Los

[16] John TheRipper es un programa de criptografía para descifrar contraseñas en estado bruto de licencia GNU GP, desarrollado por Alexander Peslyak.

ataques a la gestión de sesiones de una aplicación pueden ser utilizados para obtener acceso a cuentas de usuario sin necesidad de proporcionar credenciales correctos1*

**Pruebas para tener en cuenta a futuro:**

- Pruebas para el esquema de gestión de sesiones.
- Pruebas para atributos de cookies.
- Pruebas para fijación de sesión.
- Pruebas para variables de sesión expuestas.
- Pruebas para CSRF

### 7.2.7 Pruebas de validación de datos.

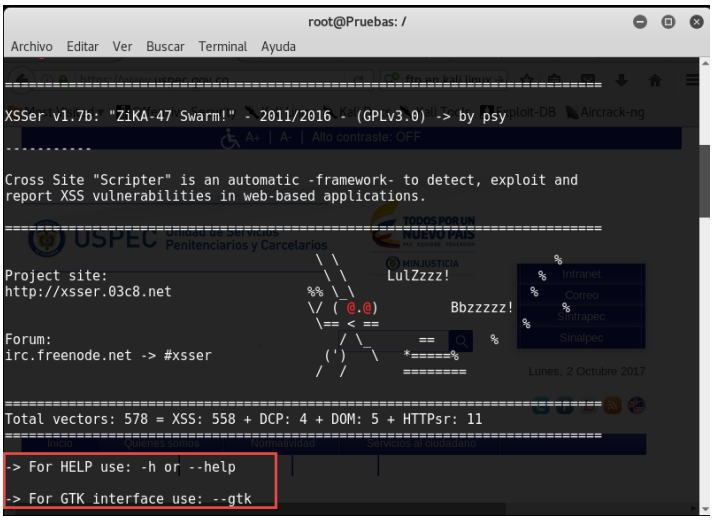
La debilidad más común en la seguridad de aplicaciones web es la falta de una validación adecuada de las entradas procedentes del cliente o del entorno de la aplicación, si no se valida los parámetros introducidos por los usuarios esto provoca que usuarios malintencionados inyectan comandos o sentencias en vez de simples datos, con el peligro que esto conlleva para la normal ejecución de la aplicación.

**Pruebas de crosssite scripting (XSS):**

Una de las formas de hacer pruebas de XSS sobre una aplicación web es usando la herramienta XSSER en la suite de Kali Linux vamos a mostrar como:

En la terminal de Kali Linux escribimos xsser, una vez hecho esto nos sale lo siguiente:

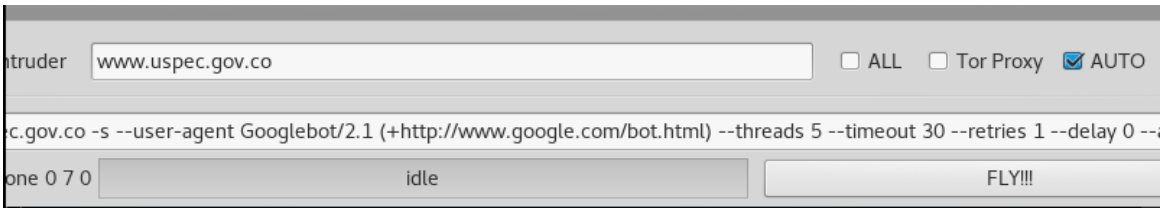
**Figura19.Comando XSSER para entorno gráfico.**




Fuente: El Autor.

Si vemos la imagen con más detalle nos indica que para usar la interfaz gráfica escribimos la siguiente orden: xsser - -gtk y nos sale lo siguiente:

**Figura20. Opciones XSSER interfaz gráfica.**



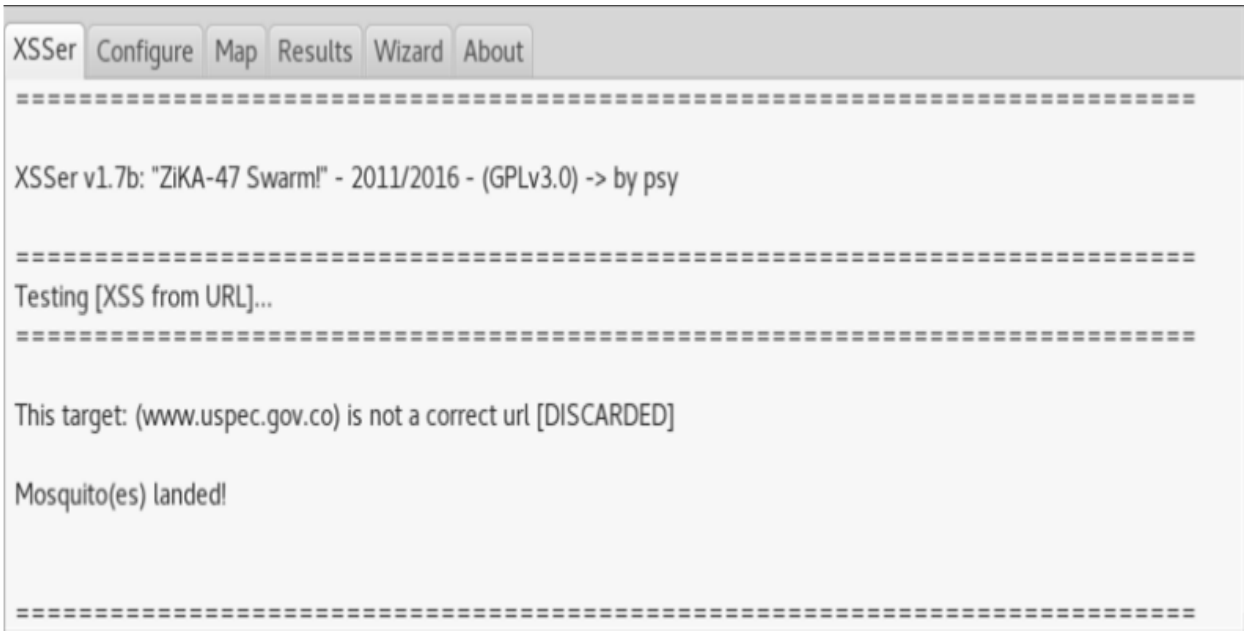

Fuente: Autor.

Donde dice flymodeexplorer o intruder son modos de exploración que contiene la herramienta, le damos clic en intruder ya que es un modo de exploración más profunda y orientada al ataque.

En el recuadro colocamos en el cuadro de texto la página web a atacar en este caso el portal web de la USPEC, le damos donde dice automatic para que automatice la labor de ataque, en el recuadro donde dice command son los

comandos predeterminados que usa la herramienta para la labor de ataque o auditoria y aparecen en ese cuadro de texto cuando le damos clic en aim, y donde dice fly y es cuando le damos clic ahí que ejecutamos la herramienta.

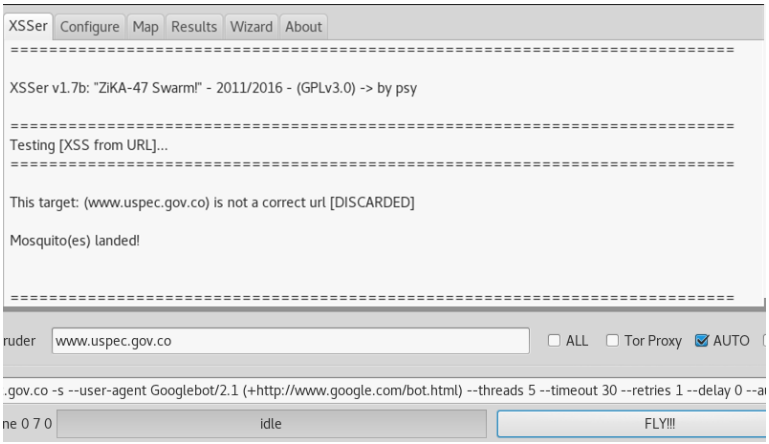
**Figura 21. Ejecución XSSER interfaz gráfica.**




Fuente: Autor.

En esta imagen aparece una dirección http modificada con algún parámetro, para probar si esta dirección modificada funciona la pegamos en la barra de dirección de cualquier navegador y si esa orden modificada o inyección funciona alterara alguna función de la página, en este caso indica el resultado de esa inyección en la página.

**Figura 22. Resultados XSSER interfaz gráfica.**




Fuente: Autor.

Podemos ver la inyección usadas en modo automático por la herramienta, vemos que no descubrió alguna vulnerabilidad en XSS, para un análisis más profundo se recomienda usar otras combinaciones en la interfaz automática o usar comandos en la terminal

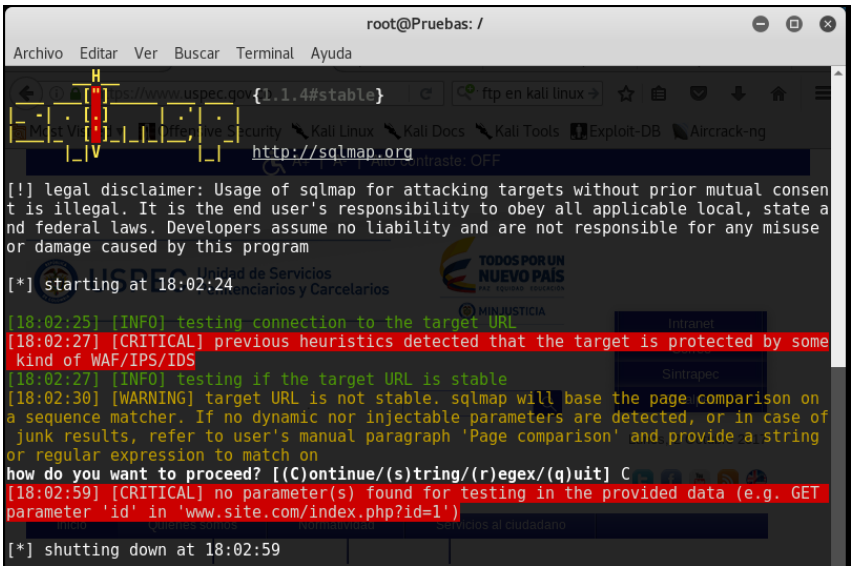
- **Inyección SQL:**

Con la herramienta SQLMAP se pretende verificar si existe esta vulnerabilidad. Una forma seria digitando la siguiente orden:

**sqlmap –u “http://www.uspec.gov.co” - -dbs**

Donde –u es la url a examinar y –dbs es para ver si busca las bases de datos asociadas a esa página.

**Figura 23.Prueba SQLMAP portal web USPEC.**




Fuente: El Autor.

En este caso vemos que el tiempo de conexión se agotó lo cual no fue posible continuar con la prueba en dado caso que hubiera tenido éxito nos debería mostrar las bases de datos que tiene esa página web, para luego probar con otras combinaciones donde podamos ver tablas, campos y demás datos que componen un Base Datos.

En caso de que hubiera tenido éxito ese comando podíamos probar lo siguiente:

**sqlmap. -u "http://www.uspec.gov.co" –D nombredeunabasededatos --tables**

**-D significa nombre de la base de datos a examinar**

**--tables significa que tablas contiene esa base de datos.**

Y si funcionara esa instrucción se hubiera profundizado de la siguiente manera:

**sqlmap. -u "http://192.168.70.22/cat.php?id=1" -D nombrebasededatos -T nombretabla --columns**

**-T significan tablas que tiene esa base de datos.**

**--columns. Muestra columnas que tiene esa tabla.**

Esto es una forma de averiguar si nuestra página web presenta problemas en la inyección de SQL, hasta ahora haciendo la exploración de la página con esta herramienta no se encontró dificultad.

### 7.2.8 Pruebas de denegación de Servicio.

En este caso se pretende lograr que la página web a auditar" no funcione ya sea saturando sus funciones en la red, memoria, su espacio de disco, entre otras perjudicando la funcionalidad y capacidades de esta.

**Pruebas para tener en cuenta a futuro:**

- Denegación de servicio mediante ataques SQL Wildcard.
- Bloqueando cuentas de usuarios.
- Desbordamiento de búfer.
- Reserva de Objetos Especificada por Usuarios.
- Pruebas de Escritura de Entradas Suministradas por Usuario a Disco.
- Fallar en la liberación de recursos.
- Pruebas de Almacenamiento Excesivo en la Sesión.

### 7.2.9 Pruebas de Servicios Web.

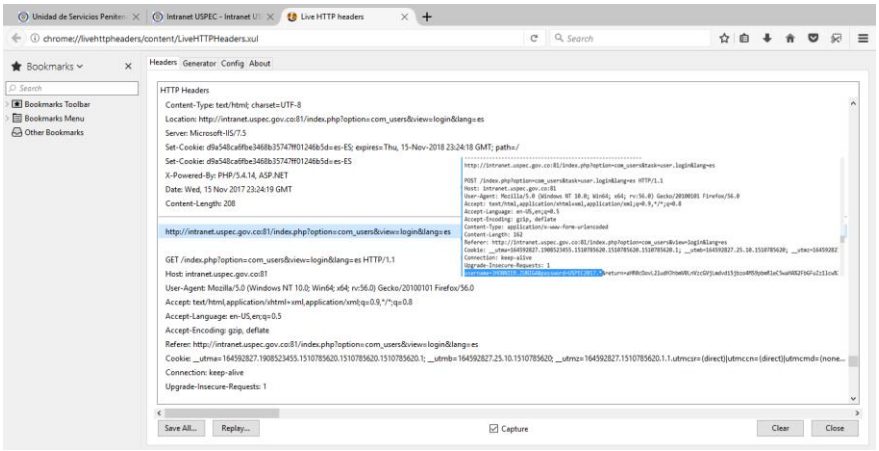
Los servicios web suelen utilizar el protocolo HTTP junto con tecnologías como XLM, SOAP, WSDL y UDDI; por lo que las pruebas de seguridad sobre servicios web deben centrarse en la búsqueda e identificación de las vulnerabilidades en dichas tecnologías[17].

[17] Expresión Binaria.Prueba de Intrusión sobre aplicaciones. Disponible en: (http://www.expresionbinaria.com/pruebas-de-intrusion-sobre-aplicaciones/).

**Comprobación de parámetros HTTP GET/REST.**

Una forma de revisar estos parámetros es con la herramienta LIVE HTTP HEADERS el cual es un addon o complemento usando en Firefox y en OWASP MANTRA para revisar los encabezados, aloja datos sobre el conjunto de caracteres, lenguaje, memoria caché, la autorización y la caducidad del contenido.

**Figura24.Uso de LIVE HTTP HEADERS.**

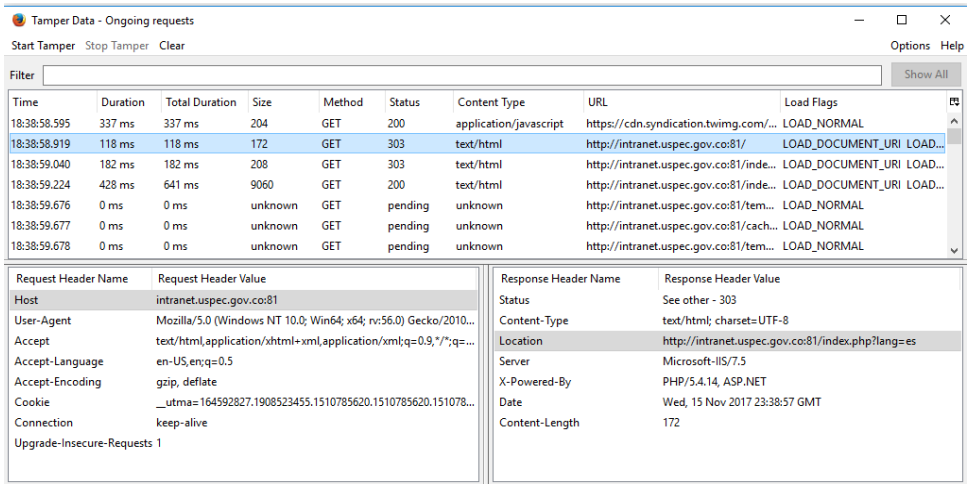



Fuente: Autor.

Una vez se abre la página este complemento comienza a capturar la información o peticiones de datos de una página como cabeceras HTTP, password y cookies que pasan por el puerto 81 de la aplicación este código se puede guardar para revisiones.

Incluso con el complemento de firefoxTamper Data nos permite modificar los encabezados HTTP y parámetros de POST, mostraremos una imagen de lo que se ve cuando se abre tamper Data

**Figura25.Uso de FIREFOX TAMPER DATA.**




Fuente: Autor.

**Pruebas para tener en cuenta a futuro:**

- Obtención de información en Servicios Web.
- Pruebas de WSDL.
- Pruebas estructurales de XML.
- Comprobación de XML a nivel de contenido
- Adjuntos SOAP maliciosos
- Pruebas de repetición

### 7.2.10 Pruebas de AJAX.

Las aplicaciones basadas en tecnologías AJAX han tenido una rápida expansión debido a la gran interactividad y facilidad de uso que proporcionan. Sin embargo, al aumentar la superficie de ataque y al procesar instrucciones tanto en el lado cliente como en el lado servidor, las vulnerabilidades de seguridad de las aplicaciones AJAX son tantas o incluso más que las de las aplicaciones desarrolladas con otras tecnologías[18]

[18]Ibíd.

**Pruebas para tener en cuenta a futuro:**

- Vulnerabilidades AJAX.
- Como probar AJAX.

Decimos que algunas de estas pruebas las tendremos a futuro ya que requieren un análisis más detallado y exhaustivo más sin embargo mostraremos algunas herramientas que nos puede facilitar la labor su funcionamiento y posibles resultados.

## 7.3 EJEMPLO DE PRUEBAS CON JOOMSCAN

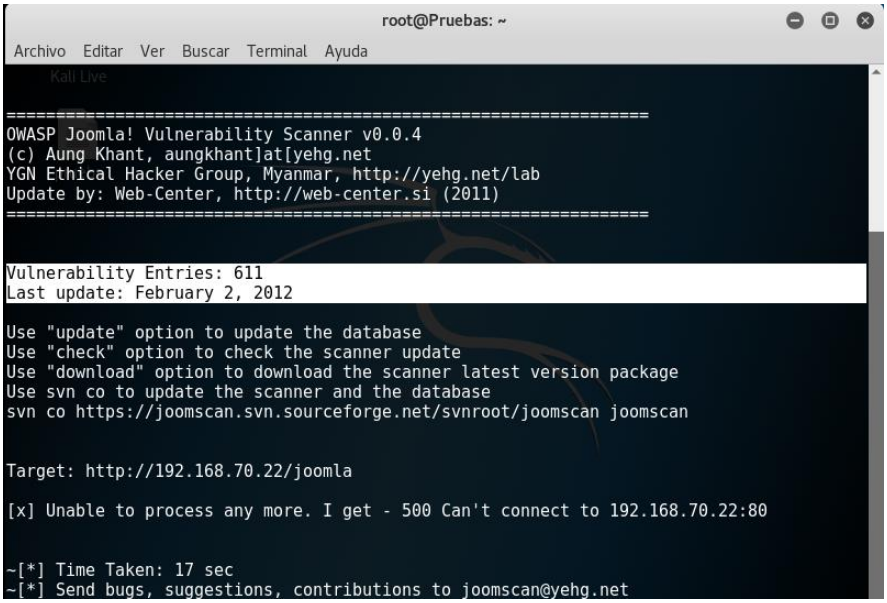
Las páginas web manejan gestor de contenidos en este caso joomla es un gestor de contenidos que facilita crear sitios web dinámicos y cambiar su contenido constantemente, lo cual hoy en día como la mayoría de las páginas son creadas por estos gestores de contenido los atacantes encuentran nuevas formas para atacar la aplicación web de un negocio basado en el uso de estas tecnologías.

Se requiere evaluar la seguridad de los plugins instalados en joomla así como su versión para esto vamos a usar la herramienta joomscan que también está en la suite de Kali Linux.

Para esto nos abrimos la terminal en Kali Linux digitamos la palabra joomscan y nos muestra las opciones de uso una vez mostradas las opciones de uso procedemos a teclear el siguiente comando:

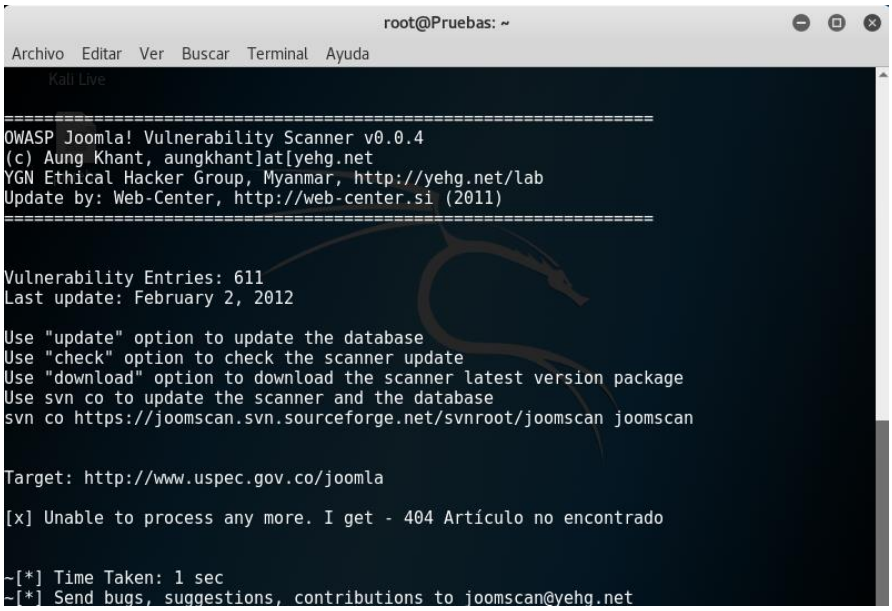
Joomscan –u www.uspec,gov.co, donde –u significa url a evaluar:

**Figura26.Interfaz de JOOMSCAN.**




Fuente: Los Autor.

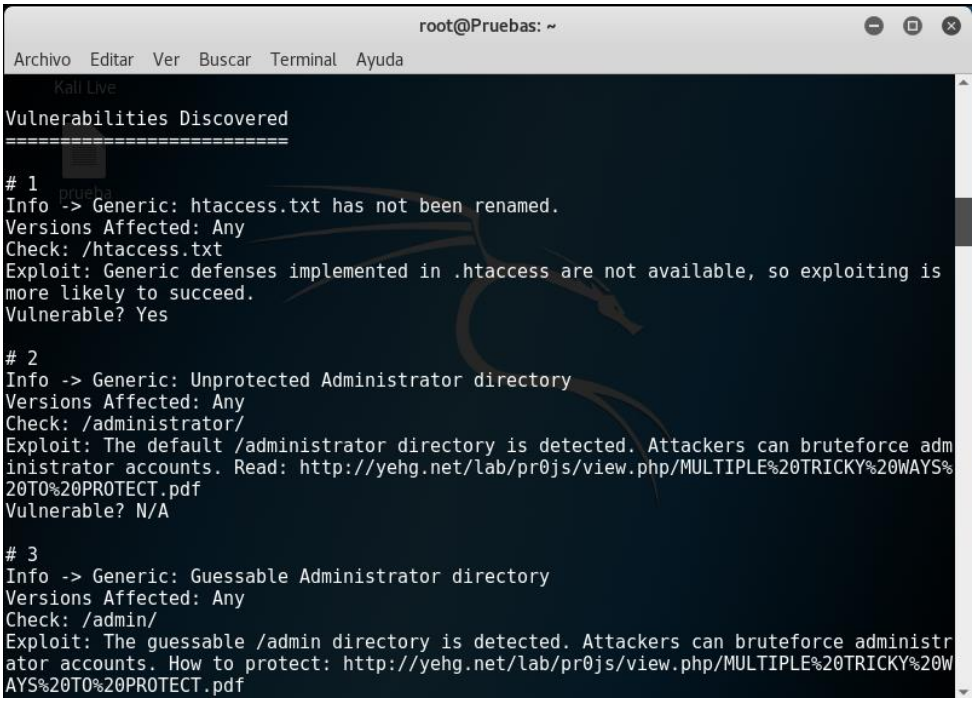
**Figura 27. JOOMSCAN en ejecución.**




Fuente: Autor

En la imagen la herramienta detectó las vulnerabilidades que aplican o no aplican en modo de progreso si es por inyección de código entre otras nos muestra las versiones desactualizadas.

**Figura 28.JOOMSCAN descubriendo vulnerabilidades.**




Fuente: Autor.

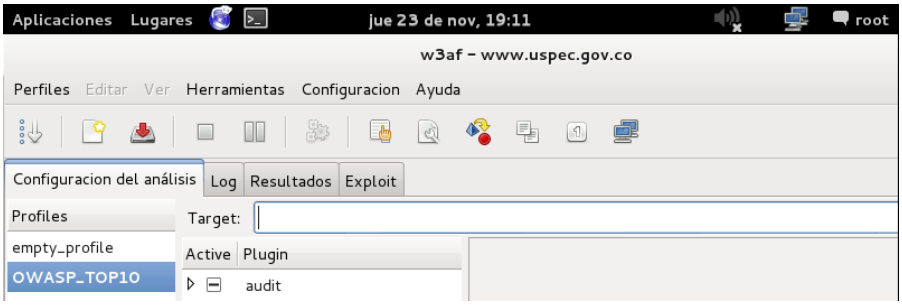
Para ver el progreso de las vulnerabilidades Joomscan –sp -u www.uspec,gov.co -oh

Ahora para ayudar en el desarrollo de escaneos y pruebas de pentest se va a ilustrar mediante dos herramientas que ayudan en la labor de descubrir vulnerabilidades y que están acorde a las pruebas de penetración del portal web USPEC proyecto OWASP.

## 7.4 EJEMPLO DE PRUEBAS HERRAMIENTA W3AF

Para usar esta herramienta se escribe en la consola de Kali Linux el comando w3af para que cargue el entorno gráfico**.** Se escribe la dirección web a auditar al lado inicia la ejecución de la aplicación W3AF, los perfiles de exploración que tiene la herramienta para exploración de vulnerabilidades , el perfil de exploración predeterminado que se eligen en este caso es OWASP_TOP10 , este perfil escanea basado en las vulnerabilidades del TOP de 10 OWASP , aunque los otros perfiles como audit_high_risk hace una auditoria buscando vulnerabilidades que nos dan alto riesgo y que afectan la página web auditada , con fast_scan se hace un escaneo más rápido de las vulnerabilidades ya que toma menos tiempo y ofrece resultados más rápido para el investigador, full_audit es una auditoria más completa, sitemap audita basado en la información contenida en el mapa de un sitio web y web_infraestructure audita la web de manera remota, bruteforce revisa mecanismo de autentificación básica y controles de acceso y full_audit_spider_man revisa la página con la metodología spider que resumen en modo texto los contenidos de un sitio web.

**Figura29.W3AF configuración de análisis.**




Fuente: Los Autor

Y lo que está en el recuadro de borde amarillo son los plugings activos que complementan los perfiles de auditoría vamos a explicar algunos de ellos:

**Audit:** Obtienen información de los plugindiscovery los cuales hacen descubrimientos para localizar vulnerabilidades como sqlinyeccion, bof, ejecución de comandos, XSS, CSRF entre otras vulnerabilidades las cuales se almacenan para su posterior posible explotación.

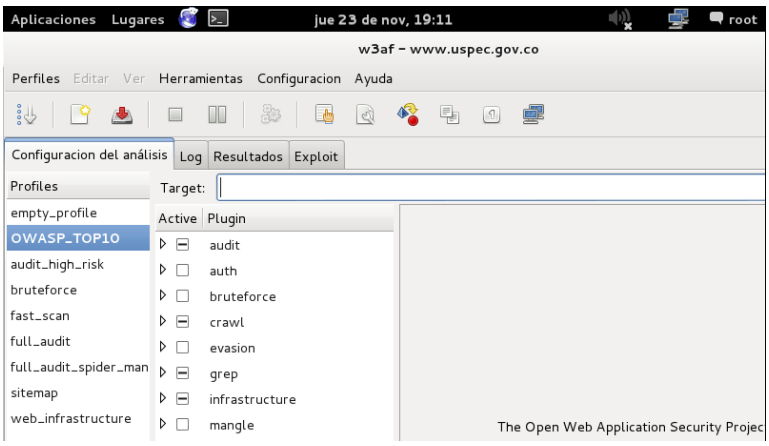
**Bruteforce:** Utilizando información recopilada de los módulos grep, podemos hacer lanzar un ataque de fuerza bruta basicAuth o de formulario[19].

**Evasión**: Nos permiten modificar las peticiones o parte de las peticiones para evadir dispositivos IDS / IPS (sistemas de detección y prevención de intrusos).

**Mangle**: Alteran peticiones y respuestas en función de expresiones regulares.

**Grep:** revisan las peticiones que vamos haciendo en busca de comentarios en código, emails, direcciones IP entre otras.

**Figura30.W3AF ejecutando perfil OWASP TOP10.**

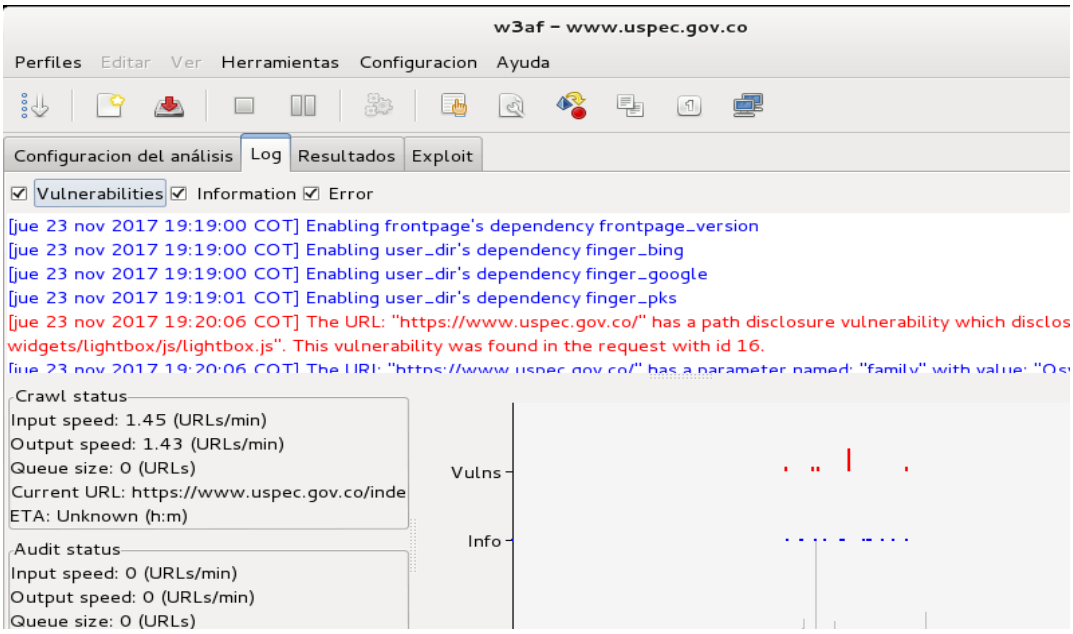



Fuente: Autor.

[19]PALANCO, José Ramón. "w3af un framework de test de intrusión web". Disponible en: (https://www.owasp.org/images/b/b5/W3af_owasp_spain_iv.pdf). Consultado el 22 de septiembre de 2015.

En la Figura 29 se evidencia al ejecutar la auditoria con el perfil OWAP_TOP10 el cual escanea teniendo en cuenta este top 10 de vulnerabilidades donde escribimos el nombre o dominio del portal web, que es la que estamos auditando.

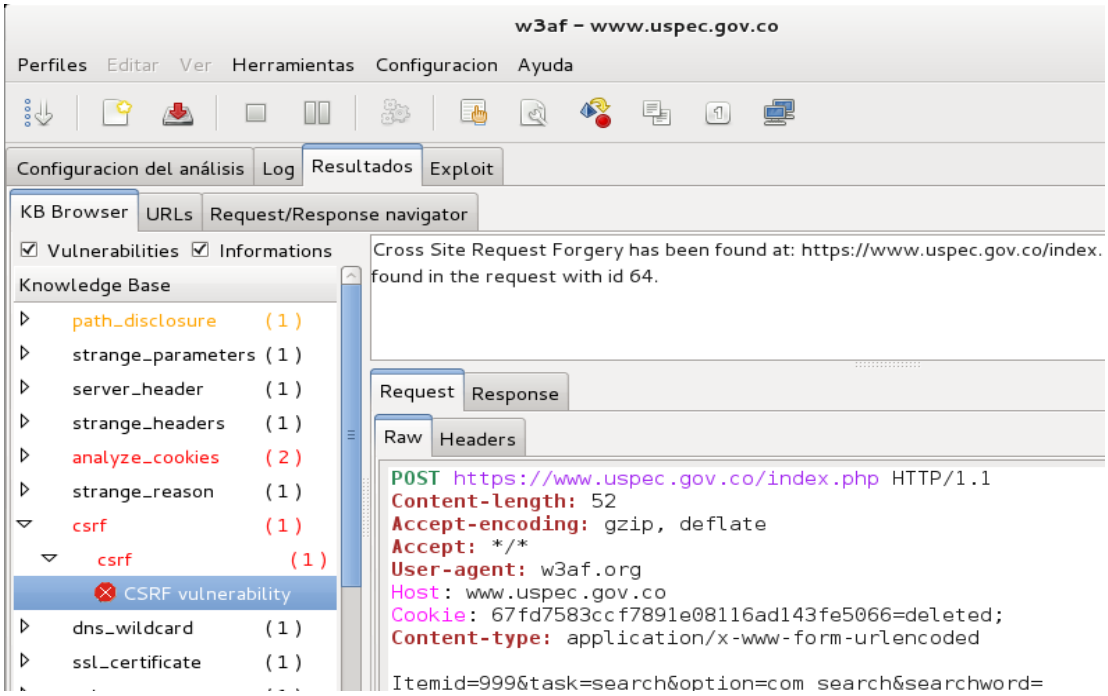
**Figura 31.Log de W3AF.**




Fuente: Autor

La pestaña log muestra las vulnerabilidades (en rojo) lo de azul es la información que va desplegando la aplicación con respecto a la página, también hay un gráfico de barra que nos muestra la evolución del escaneo.

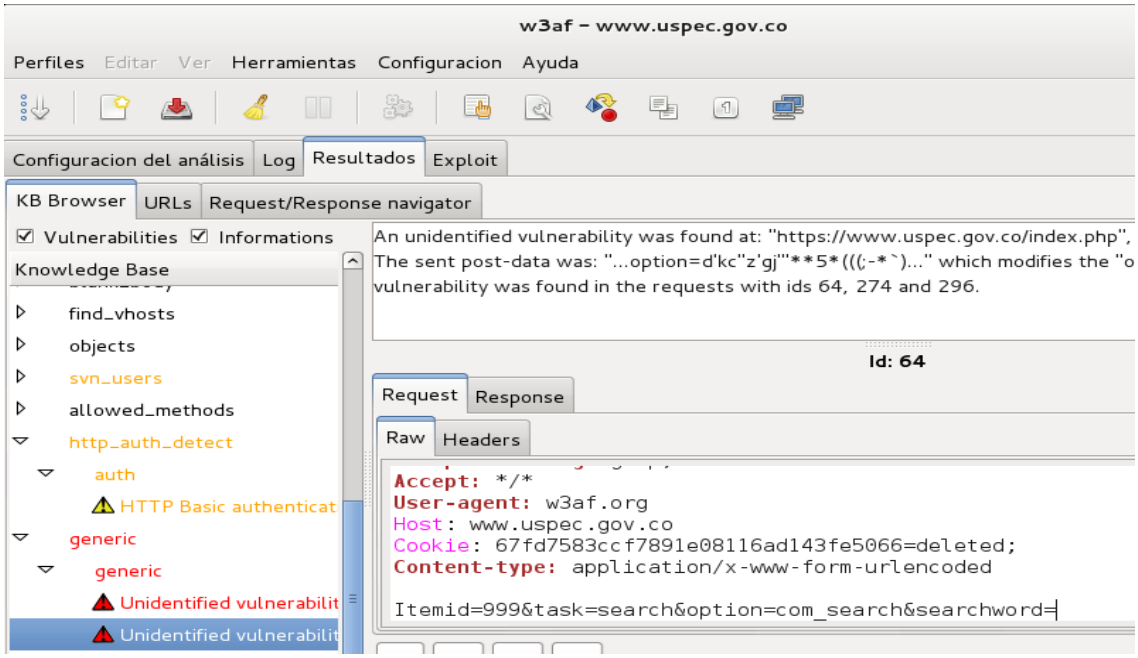
**Figura 32. Detección de vulnerabilidad CSRF en W3AF.**




Fuente: Autor

Muestra información y vulnerabilidades encontradas, en este caso encontró una vulnerabilidad CSRF, muestra un mensaje de la vulnerabilidad encontrada con su código, si vemos la pestaña Request en la pestaña raw se observa la cookie donde se presenta este problema, si le damos a la pestaña headers nos mostrara el encabezado donde se presenta esta vulnerabilidad.

**Figura 33. Detección de vulnerabilidad Generic y HTTP Basic en W3AF.**

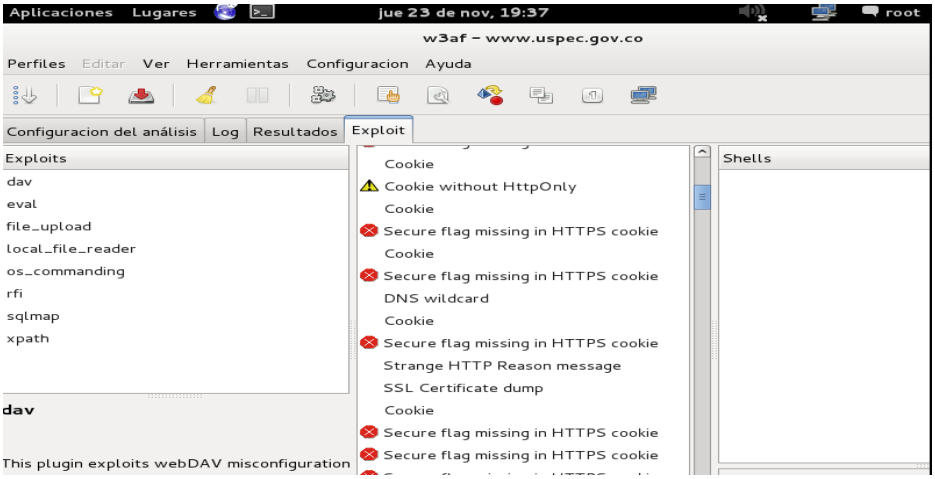



Fuente: Autor

En rojo observamos una vulnerabilidad GenericUnidentifiedVulnerability se puede eliminar contenido de la página y HTTP Basic Authenticate muestra que tiene una baja autenticidad, algo grave puede acarrear cambios en el portal[20] y podemos ver sus características revisando las pestañas Request y el mensaje que presenta la aplicación.

[20]RODRÍGUEZ, Ricardo Martín.Algunos ejemplos y defensas contra el clickjacking". Disponible en (http://blog.elevenpaths.com/2013/10/algunos-ejemplos-y-defensas-contra-el.html). Consultado el 22 de septiembre de 2015.

**Figura 34. Pestaña Exploit en W3AF.**

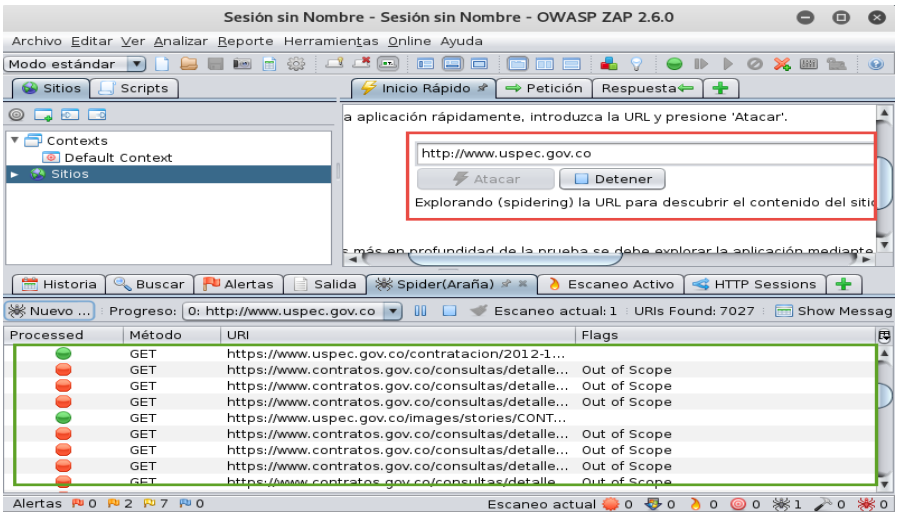



Fuente: Autor

La Figura 33 esta pestaña exploit sirve para cargar exploit predeterminados y aplicárselos a alguna vulnerabilidad para su posterior análisis, aunque por ahora se quiere mostrar que vulnerabilidades se encontraron.

## 7.5 EJEMPLO DE PRUEBAS CON OWASP ZAP

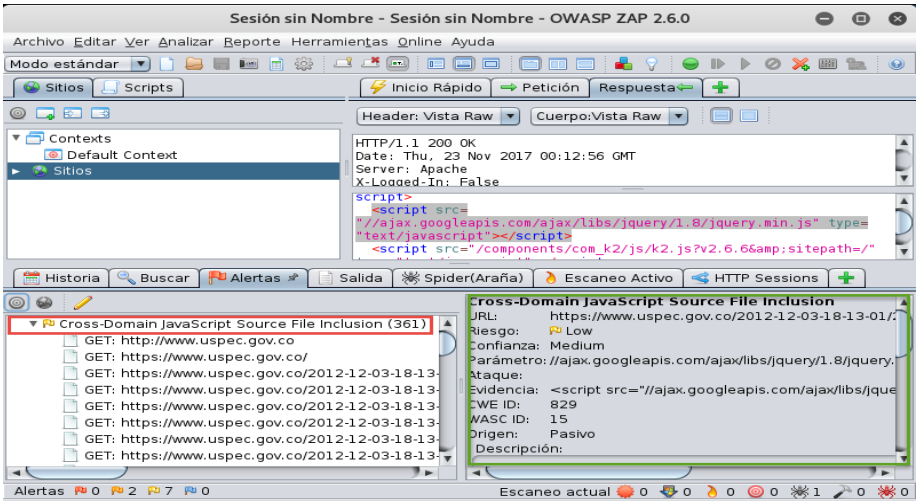
**Figura 35. Análisis del portal web USPEC con OWASP-ZAP.**




Fuente: Autor

La Figura 34 donde está el recuadro de borde rojo después de haber atacado la página muestra el riesgo, parámetros y evidencia los scripts que en un escenario como este se pueden incluir archivos, el recuadro de borde rojo en alertas muestra que tipo métodos genera el escaneo del portal web USPEC.

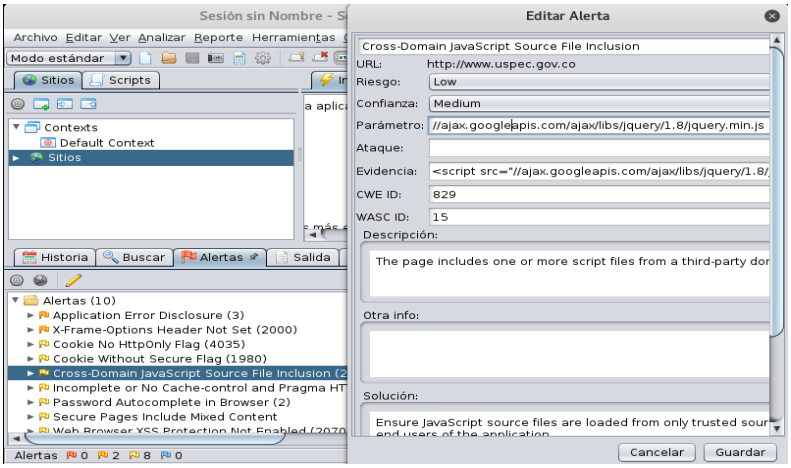
**Figura 36.Vulnerabilidades web USPEC con OWASP-ZAP.**




Fuente: Autor

La Figura 35 donde está el recuadro de borde verde después de haber atacado la página muestra el riesgo, parámetros y evidencia los scripts que en un escenario como este un podría incluir archivos, si vemos el recuadro de borde rojo en alertas vemos que tipo de alerta genera el escaneo del portal web USPEC.

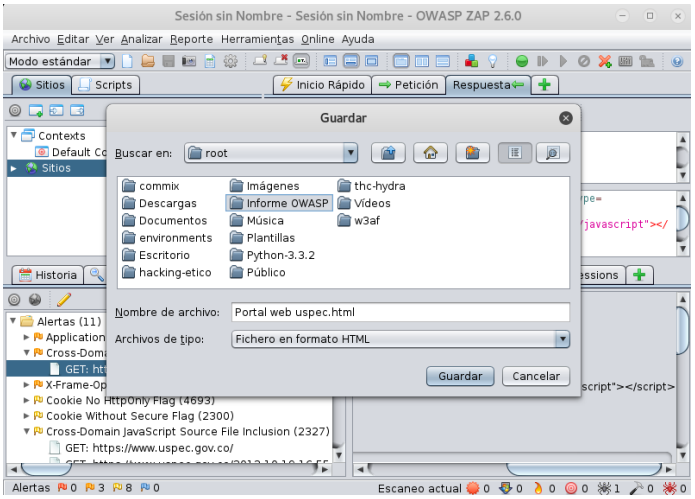
**Figura 37.Información vulnerabilidad específica.**




Fuente: Los Autores

La Figura 36 muestra las alertas doble clic a alguna vulnerabilidad específica, y muestra en el recuadro llamado editar alerta donde se ve la información de una vulnerabilidad en detalle y posible solución.

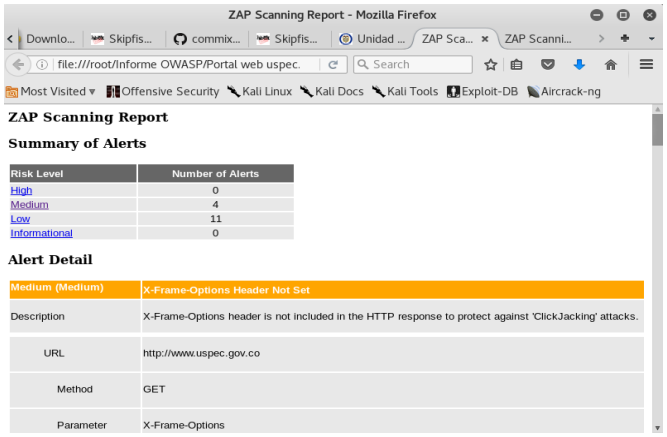
**Figura 38.Guardar informe de resultados OWASP-ZAP**.




Fuente: Autor

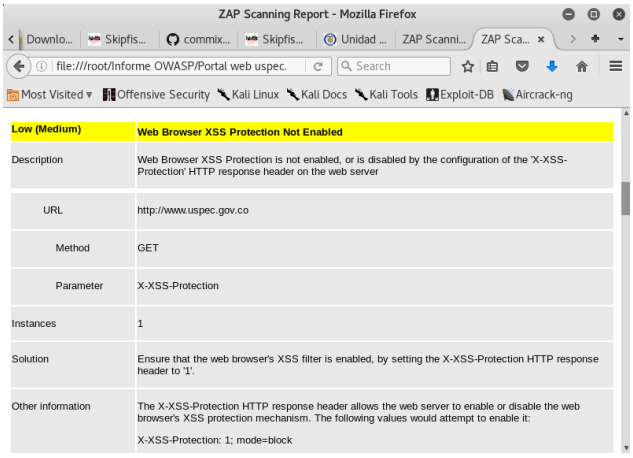
Ahora observando la Figura 38 se puede obtener un informe de todas las vulnerabilidades le damos clic a informes en la parte superior de la aplicación, se puede guardar en formato HTML o página web en alguna ubicación.

**Figura 39. Informe de resultados OWASP-ZAP.**




Fuente: Autor

**Figura 40. Informe de resultados OWASP-ZAP.**




Fuente: Autor

En la Figura 39 el informe se guardó en formato HTML. Y con esta información se puede obtener datos vulnerables los cuales son el punto de partida para hallar alguna solución a activos del portal web USPEC.

## 7.6 METODOLOGÍA DE PRUEBAS

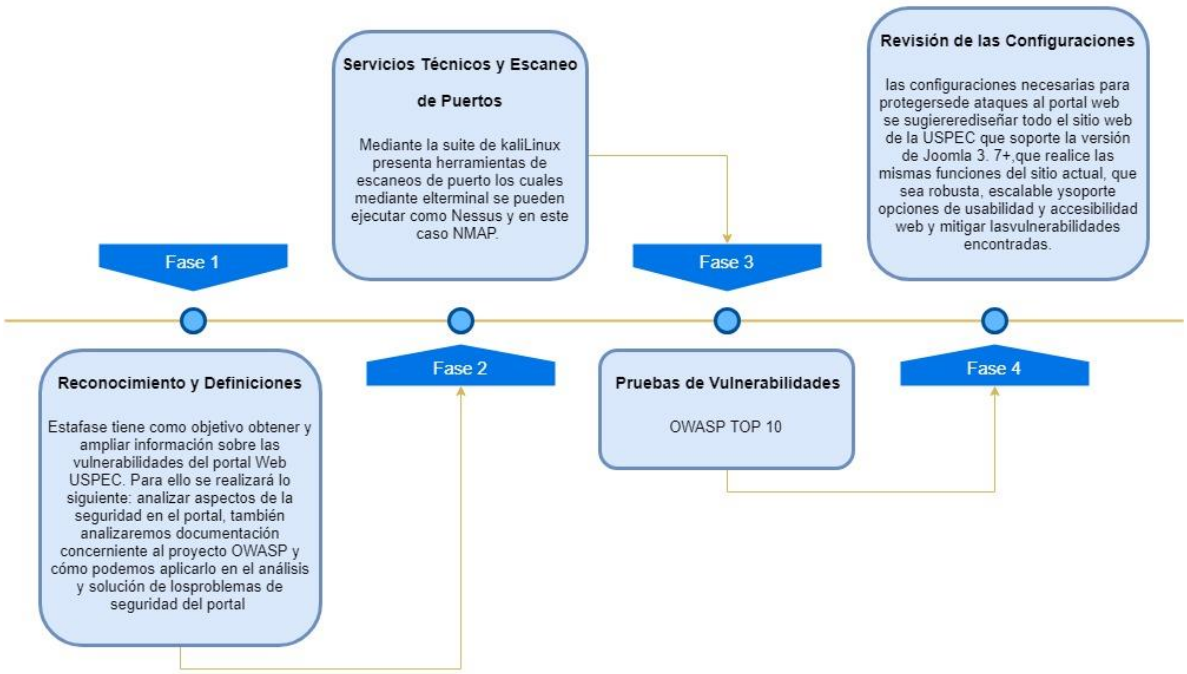
Este proyecto presenta una metodología para gestionar la auditoria de seguridad del portal web de la unidad penitenciaria y carcelaria USPEC basado en el proyecto OWASP que identifica los principales riesgos de seguridad en aplicaciones y ofrece soluciones para las mejores prácticas. El portal web por su contenido informativo además de manejar proyectos, y publicaciones de los servicios a los requerimientos y/o servicios de bienes y financieros para las reclusiones a nivel nacional como para usuarios internos y externos de la unidad. Se facilita que al poseer un portal web desactualizado comiencen a ocurrir errores y desviaciones que pueden comprometer la propia subsistencia del portal, Esta realidad causa la necesidad de realizar procesos de auditoría del funcionamiento que permitan detectar problemas graves de vulnerabilidad, establecer políticas, realizar actualización de su sistema de contenidos e incluso detectar problemas de programación que logran poner en riesgo la operación futura del portal web USPEC.

Durante los procesos de auditoría Pent Test es trascendental, garantizar que las consecuencias sean correctos y completas, para obtener un resultado uniforme, reduciendo la importancia de los niveles de técnica, instrucción, audacia, conocimiento del portal web auditado.

La metodología para la detección de vulnerabilidades en el portal web expuestas en este proyecto PENTESTING PARA EL PORTAL WEB DE LA USPEC, APOYADO EN EL PROYECTO DE SEGURIDAD OWASP, Consta de cuatro fases probadas con las herramientas del software libre Kali, mediante las cuales se

busca obtener las vulnerabilidades sobre el portal web USPEC. Esta metodología se soporta cada etapa en herramientas software mediante el proyecto OWASP. Por lo que en cada fase se puntualizan las acciones que se deben realizar y cómo se deben llevar a cabo a través de las herramientas apropiadas en 4 fases:

**Figura 41.Fases metodología.**




Fuente: Autor

- Reconocimiento y Definiciones.

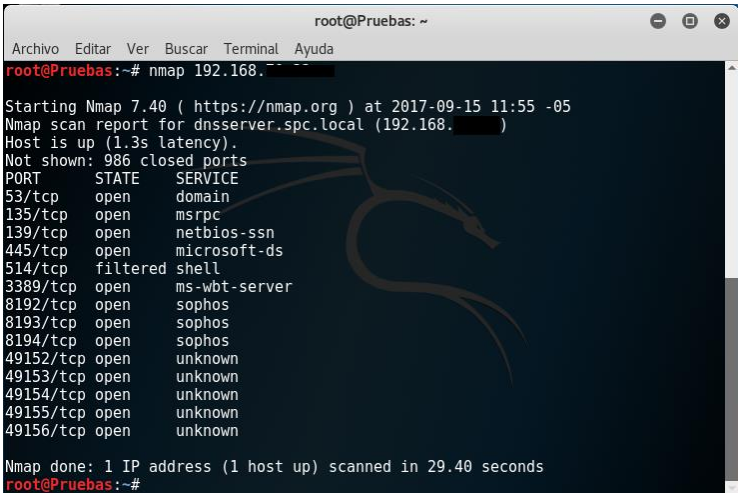
Esta fase tiene como objetivo obtener y ampliar información sobre las vulnerabilidades del portal Web USPEC. Principalmente se pretende ampliar la información con detalles prestados por los administradores del portal quien por utilizar CMS obsoleto y no poder realizar actualizaciones de los pluguines encuentran que la pagina no se puede actualizar por su sistema de contenidos no son compatibles. Para ello se realizará lo siguiente: analizar aspectos de la seguridad en el portal, también analizaremos documentación concerniente al proyecto OWASP y cómo podemos aplicarlo en el análisis y solución de los

problemas de seguridad del portal, herramientas del software Kali Linux fuerza bruta, transferencias de zona, escaneos de puertos, tipos de sistemas, contenidos, servicio asociados inspección de código certificado de seguridad cabeceras Detección de vulnerabilidad CSRF en W3AF. Cabe resaltar que en esta fase no se busca corregir ninguna vulnerabilidad en absoluto, lo que se pretende es obtener la mayor cantidad posible de fallas que podamos corregir mediante el apoyo de OWASP.

- Servicios técnicos y Escaneo de puertos

Mediante la suite de kali Linux presenta herramientas de escaneos de puerto los cuales mediante el terminal se pueden ejecutar como Nessus y en este caso NMAP.

**Figura 42.Escaneo puertos**




Fuente: Autor

Pruebas de Vulnerabilidades

Pruebas de gestión de la configuración de la aplicación.

Pruebas de Autenticación.

Pruebas de fuerza bruta.

Pruebas de Autorización.

Pruebas de gestión de sesiones.

Pruebas de validación de datos.

Pruebas de denegación de Servicio.

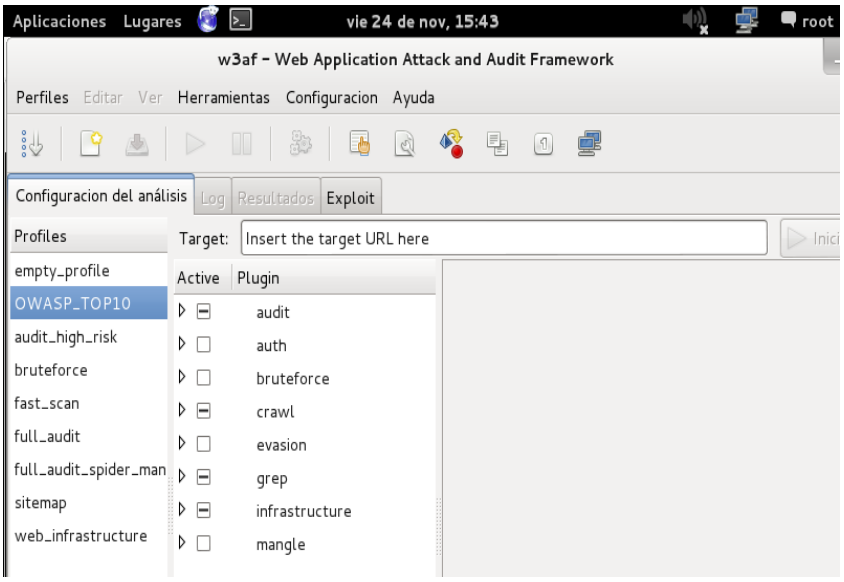
Pruebas de Servicios Web.

Pruebas de AJAX.

- Revisiones de las configuraciones.

Se realizaron configuraciones con el OWASP TOP 10 de la Suite de Kali.

**Figura 43.Configuraciones OWASP**




Fuente: Autor

- Resultados y Recomendaciones

El modelo fue sometido a una validación. La arquitectura de TI y de seguridad de la dependencia que se utilizó para realizar la prueba del modelo fue modificada para atender las recomendaciones emanadas de la utilización que se propone en este trabajo. La utilización del modelo no sólo permitió detectar condiciones de riesgo en la plataforma ya existente, sino también detectar insuficiencias

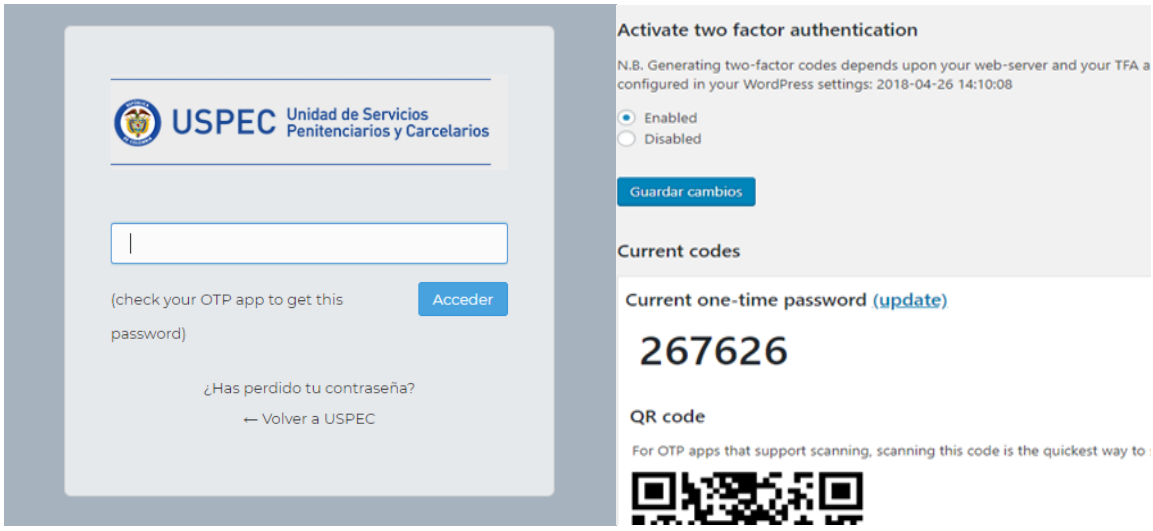
importantes en el dimensionamiento del equipamiento y carencias en temas vitales como las Políticas de Seguridad Informática.

Con las soluciones brindadas para el portal web de la unidad de servicios penitenciarios y carcelarios actualizando el CMS por una versión más segura a la cual se le instalaron los siguientes plugin de seguridad para mejorar la estabilidad del portal web

Two Factor Authentication Configuración:

Genera códigos de dos factores de autenticación, de la aplicación / dispositivo de TFA que se acuerde en el momento, en este caso a cada usuario que tiene acceso a la administración del gestor deberá ingresar a través de una app llamada autenticator de google un código aleatorio que se denomina toquen para el debido ingreso y en dado caso que se ingrese mal, el complemento.

**Figura 44.Two Factor Authentication**



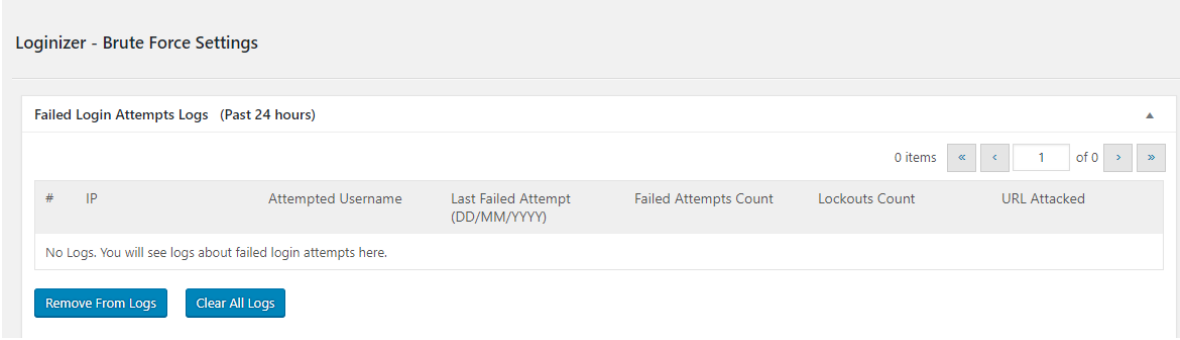

Fuente: Autor

Se instala plugins llamado Two factor authentication, El cual funciona perfectamente para la doble capa de seguridad en el acceso al panel de administración wp-admin

LoginizerDashboard lo reportara y en dado caso en 3 intentos fallidos lo bloqueara

Seguridad en el Inicio de sesión. Por defecto, BruteForceProtection se habilita de inmediato y la configuración está ajustada según las necesidades.

**Figura 45.Plugins para prevenir ataques de fuerza bruta**




Fuente: Autor

Además también se instala plugins para prevenir ataques de fuerza bruta, llamado loginizerDashboard

Wordfence Security – Firewall & Malware Scan:

Asegura el sitio, este plugins de seguridad en WordPress es el más completo. En cuanto a Cortafuegos, escaneo de malware, bloqueos, tráfico en directo, seguridad de acceso y más.

# 8 RECURSOS DISPONIBLES

**Tabla 1. Recursos disponibles para el desarrollo del proyecto**

| RECURSOS HUMANOS | CATEGORÍA CIENTÍFICA, DOCENTE O TECNOLÓGICA | INSTITUCIÓN A LA QUE PERTENECE | HORAS DESTINADAS AL PROYECTO | $/HORA | TOTAL |
|---|---|---|---|---|---|
| **Nombre de los demás participantes (Asesor)** Edgar Alonso Bojacá | Magister | UNAD | Entre 16 Horas | 45.000 | 1.920.000 |
| **Nombre del Estudiante** Jhonnier Zuñiga Mosquera | Ingeniería | UNAD | Entre 40 Horas | 30.000 | 1.200.000 |
| *TOTAL, RECURSOS HUMANOS: 3.120.000 CO* | | | | | |

Fuente: Autor

**Tabla 2. Recursos Materiales del proyecto**

| RECURSOS MATERIALES Y OTROS RECURSOS | COSTO | FUENTE DE FINANCIAMIENTO |
|---|---|---|
| Recursos materiales | | |
| 1 equipos de escritorio | 2.000.000 | Recursos propios equipo investigador |
| 1 portátiles | 1.800.000 | Recursos propios |

| | | |
|---|---|---|
| | | equipo investigador |
| TOTAL, RECURSOS MATERIALES | **3.800.000** | Recursos propios equipo investigador |

Fuente: Autor

**Tabla 3. Recursos de infraestructura tecnológica del proyecto**

| **RECURSOS MATERIALES Y OTROS RECURSOS** | **COSTO** | **FUENTE DE FINANCIAMIENTO** |
|---|---|---|
| Recursos materiales | | |
| Certificados de seguridad | **1.750.000** | USPEC |
| Dominio | **Spc.local** | USPEC |
| Configuración y Soporte | **Ingenieros área tecnología** | USPEC |
| Servidores | **USPEC** | USPEC |
| TOTAL, RECURSOS MATERIALES | | USPEC |

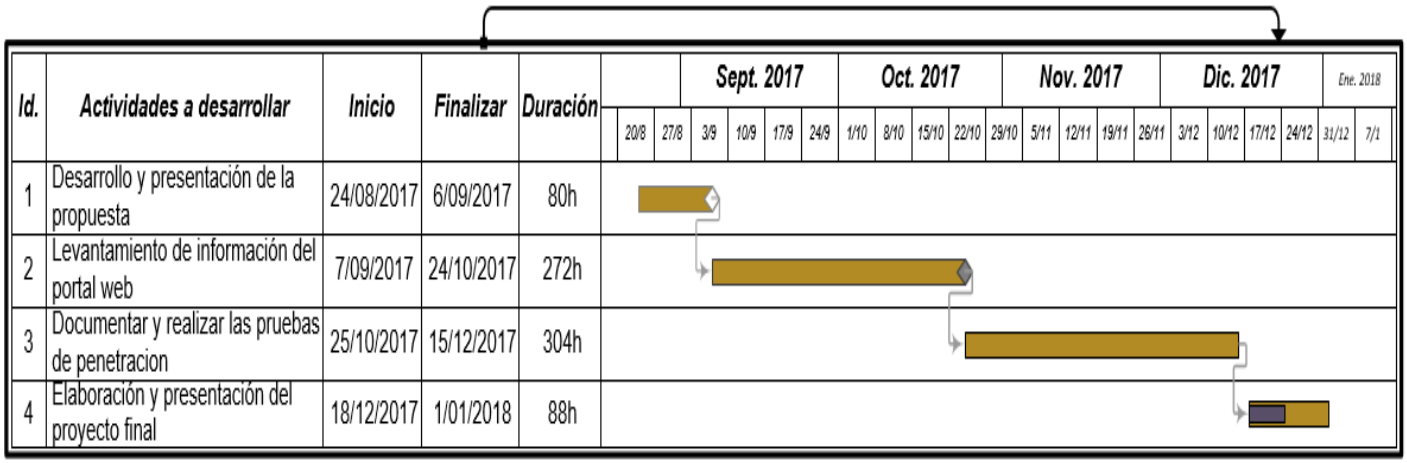
Fuente: Autor

**Tabla 4. Recursos de software para aplicación del proyecto**

| RECURSOS MATERIALES Y OTROS RECURSOS | COSTO | FUENTE DE FINANCIAMIENTO |
|---|---|---|
| Recursos materiales | | |
| **1) Herramientas de gestión de pruebas**<br>BugzillaTestopia<br>qaBook<br>RTH (open source)<br>Radi-testdir<br>Data Generator<br>Suite de Kalil Linux | **Free** | **Ninguna** |
| **2) Herramientas para pruebas funcionales**<br>Watir WatiN Capedit<br>ITP<br>WET<br>WebInject | **Free** | **Ninguna** |
| **3) Herramientas para pruebas de carga y rendimiento**<br>FunkLoad<br>FWPTT load testing<br>loadUI | **Free** | **Ninguna** |
| 4) CMS Joomla | **Licencia Institucional** | Recursos propios de la institución |
| 5) Dreamweaver | **Licencia Institucional** | Recursos propios de la institución |
| 6) Herramientas de desarrollo de bases de datos<br>MySQL<br>PHP 5.3 | **Licencia Institucional** | Recursos propios de la institución |
| Subtotal | | |
| TOTAL, RECURSOS MATERIALES | | |

Fuente: http://sentidoweb.com/2006/10/20/lista-de-herramientas-para-testeo-de-aplicaciones-web.php

# 9 CRONOGRAMA



| Id. | Actividades a desarrollar | Inicio | Finalizar | Duración |
|---|---|---|---|---|
| 1 | Desarrollo y presentación de la propuesta | 24/08/2017 | 6/09/2017 | 80h |
| 2 | Levantamiento de información del portal web | 7/09/2017 | 24/10/2017 | 272h |
| 3 | Documentar y realizar las pruebas de penetracion | 25/10/2017 | 15/12/2017 | 304h |
| 4 | Elaboración y presentación del proyecto final | 18/12/2017 | 1/01/2018 | 88h |

# 10 RESULTADOS

Las pruebas con diversas herramientas software incluidas en la suite de Kali Linux detallan información relevante en el portal web como su dirección IP, gestor de contenido a usado ,conexiones abiertas , sistema operativo usado en el host o servidores, servicios de software usados por algunos puertos entre otras, esta información es relevante ya que algún potencial atacante la puede usar para hacer un ataque más efectivo a esta página, y todo esto se puede hacer efectivo siempre y cuando el atacante tenga conocimientos avanzados y cuente con documentación y uso de herramientas automatizadas como las que están en la suite de Kali Linux .

Mediante a los problemas identificados durante los escaneos y pruebas de penetración al portal web de la unidad de servicios carcelarios y penitenciarios se han identificado una serie de vulnerabilidades en las que se debería trabajar con tal mejorar la seguridad para los activos de información establecidos y publicados en el portal web que son de mirar con importancia por ser una entidad que se dedica al manejo jurídico y administrativo del sistema penitenciario y carcelario a nivel nacional, la información que se maneja no sólo es de carácter autentico y legítimo y se encuentra por ley bajo protección especial del Estado Colombiano ya que se trata de todo lo que concierne a la Población Privada de La Libertad y a la salvaguarda del Sistema Carcelario del País.

La página web actual está desarrollada en un gestor de contenido llamado Joomla Versión 2.5 tiene problemas de vulnerabilidad y está quedando obsoleta ya que la última versión actual de Joomla es 3.7, es decir han pasado muchas versiones y no se ha hecho la respectiva actualización y ahora para hacerla necesitamos comenzarla desde cero ya que existen plugins implementados en la anterior versión que en la actualidad son muy diferentes.

Como primera medida para el portal web de uspec se le instalaron los certificados de seguridad para prevenir vulnerabilidades de tipo SSL conjunto con el área de tecnología.

Vulnerably: X.509 Certificate Subject CN Does Not Match the Entity Name

Nombre del servicio: HTTPS

Puerto: 443

Dirección IP: 190.60.111.97

Descripción: Antes de emitir un certificado, una Autoridad de Certificación (CA) se verifica la identidad de la entidad que solicita el certificado. Por lo tanto, los procedimientos estándar de validación de certificados requieren que el campo CN sujeto de un certificado coincida con el nombre real de la entidad que presenta el certificado. Por ejemplo, en un certificado presentado por "https://www.example.com/", el CN debe ser "www.example.com".

Para detectar y evitar ataques de espionaje activo, se debe verificar la validez de un certificado o, de lo contrario, un atacante podría lanzar un ataque y obtener el control total de la secuencia de datos.

Solución: El campo de nombre común (CN) del sujeto en el certificado X.509 se debe fijar para reflejar el nombre de la entidad que presenta el certificado (por ejemplo, el nombre de host). Esto se hace generando un nuevo certificado generalmente firmado por una Autoridad de Certificación (CA) en la que confían tanto el cliente como el servidor.

Otra medida que se recomienda es la actualización de las bases de datos a su última versión. El Objetivo de esta es cambiar versión del gestor de bases de datos Mysql a su versión 5.6. para mejorar el nivel de seguridad en las bases de datos y mitigar las vulnerabilidades debido a que ya las que usan son obsoletas.

**Version de Mysql: 5.6**

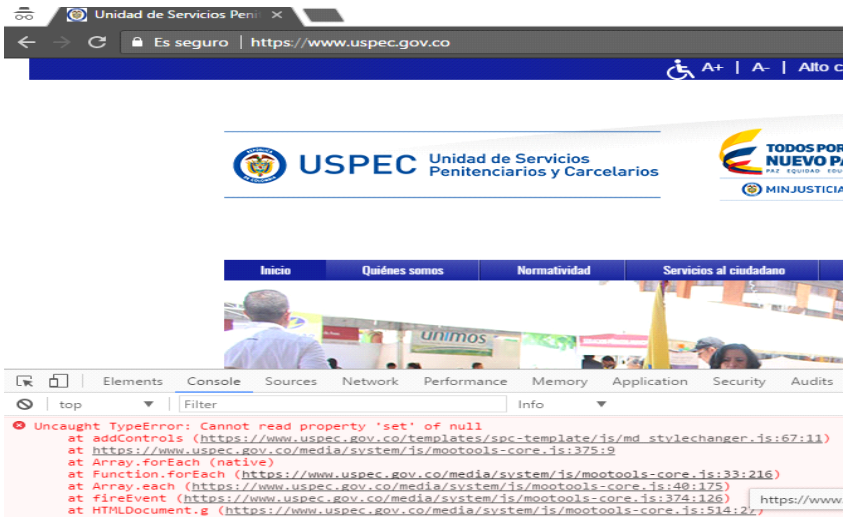
**Version actual Joomla: 2.5**

**Version de PHP: 5.3**

- **Estado de los templates página web:**

Los templates están generando errores, no son robustos y escalables. Este template se modificó a partir de una plantilla base, no está diseñado para soportar mejoras futuras, cualquier cambio que se haga generaría nuevos errores en el portal. Se sugiere rediseñarla con los mismos estilos de página web de la USPEC y que soporte la versión de Joomla 3.7+.

**Figura 46. Informe de resultados Certificados SSL/TLS.**

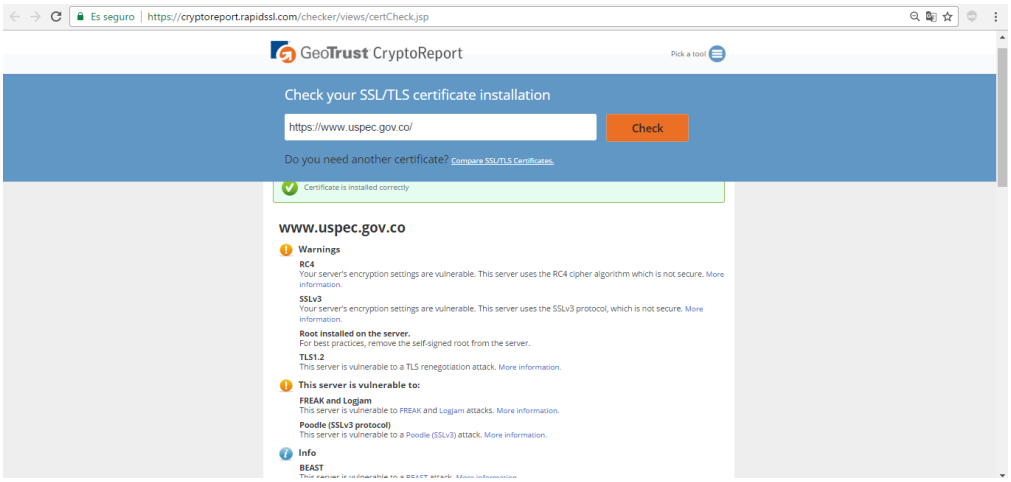



Fuente: Autor

- **Estado de las extensiones:**

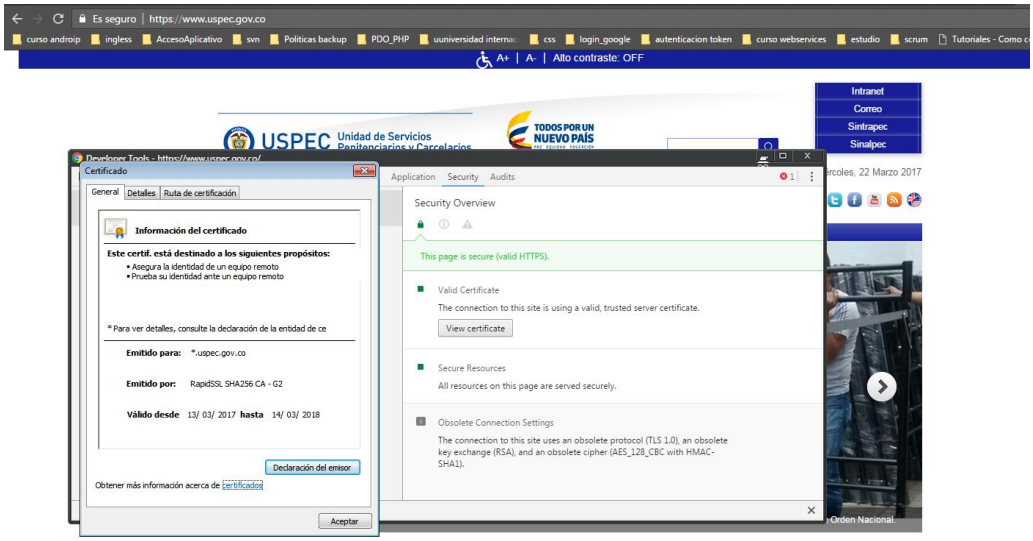
Las extensiones actualmente están generando errores en el portal, como por ejemplo el de galería de imágenes. Además, algunas extensiones no tienen actualizaciones son versiones obsoletas e inestables. Se sugiere instalar nuevas extensiones que realicen las mismas funciones de las versiones previas y que soporten la versión de Joomla 3.7+.

**Figura 47. Informe de resultados Certificados SSL/TLS.**




Fuente: Autor

**Figura 48Certificados SSL/TLS.**



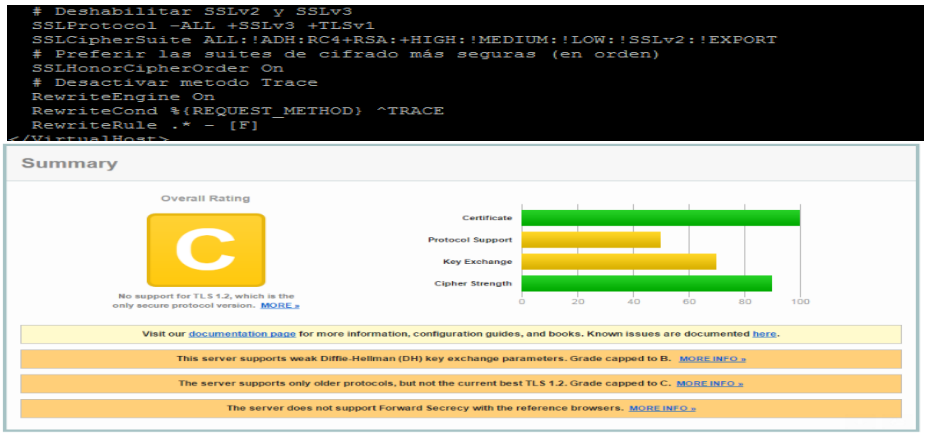

Fuente: Autor

Verificar tipo de cifrado sitio web SSL

https://www.ssllabs.com/ssltest/analyze.html?d=www.uspec.gov.co

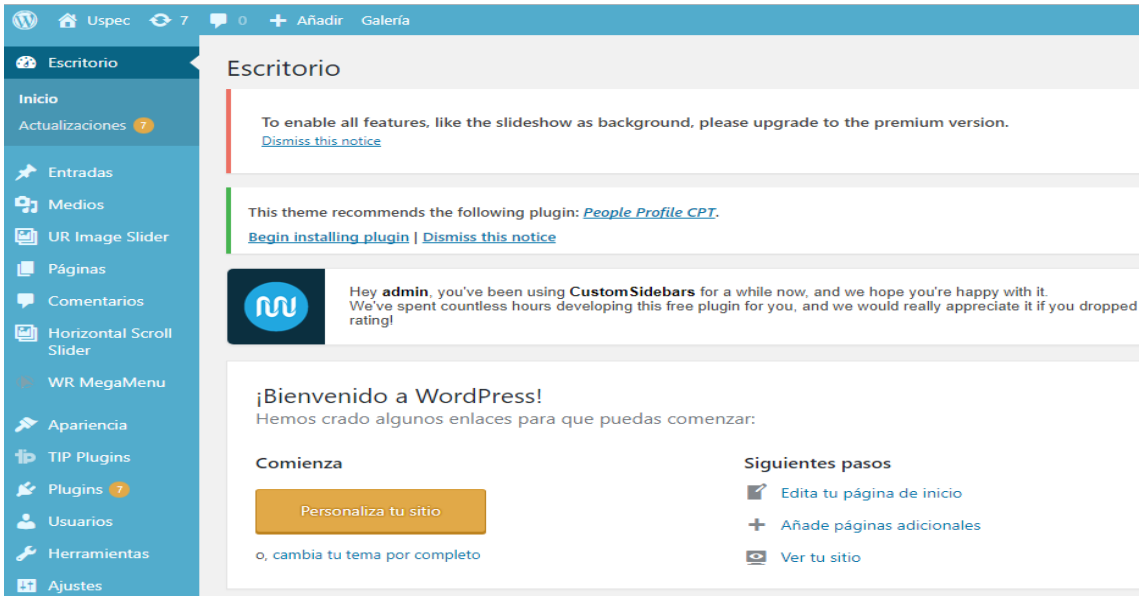
Habilitar TLS 1.1 en el servidor de la página web de la USPEC

**Figura 49. Tipo de Cifrado.**



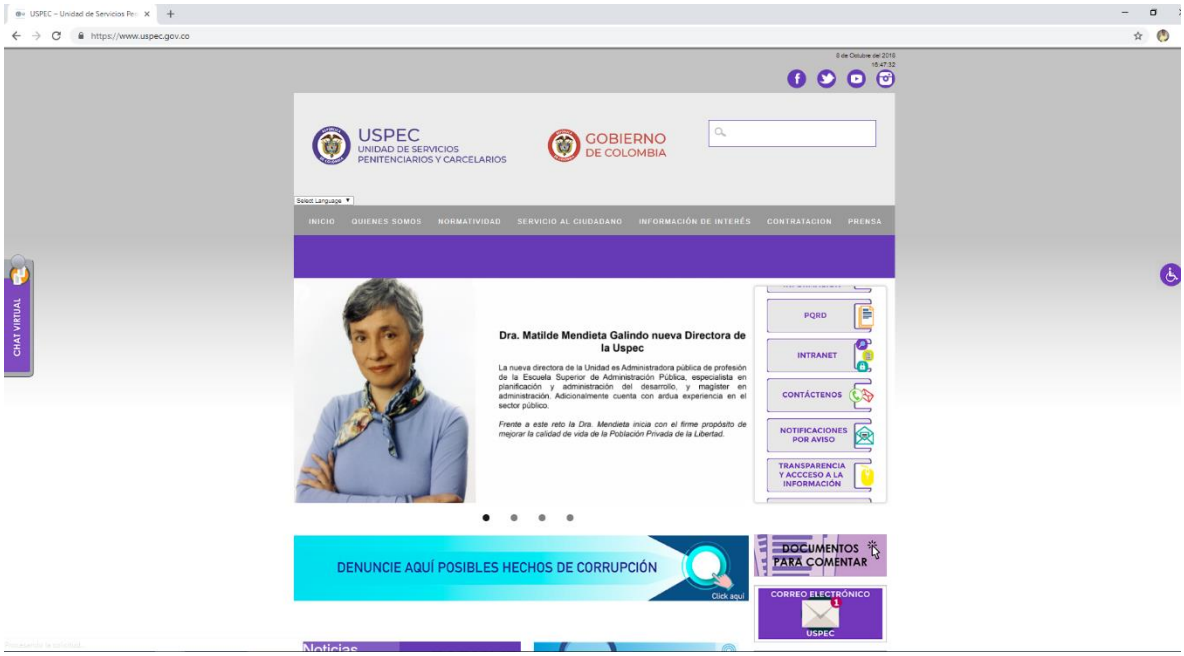

Resultado de los avances en la actualización y desarrollo del portal web de la Uspec después de haber realizado las pruebas de penetración y escaneos los cuales detallaron que por el estado obsoleto del gestor de contenidos es viable el rediseño del portal que corrija las vulnerabilidades ilustradas.

**Figura 50.Desarrollo de la Página en Gestor de contenido WordPress.**




Fuente: Autor

**Figura 51.Avance del Nuevo home del portal desarrollado en Wordpress**




Fuente: Autor

# 11 RECOMENDACIONES

Realizar las configuraciones necesarias para protegerse de ataques al portal web se sugiere rediseñar todo el sitio web de la USPEC que soporte la versión de Joomla 3. 7+, que realice las mismas funciones del sitio actual, que sea robusta, escalable y soporte opciones de usabilidad y accesibilidad web y mitigar las vulnerabilidades encontradas.

En la actualidad existen otros gestores de contenidos, los cuales se recomienda uno bastante utilizado por grandes empresas y que corrige una gran parte de las vulnerabilidades es WordPress como se observa en la Figura 44.

Es un software de código abierto desarrollado de manera espléndida por miles de desarrolladores que colaboran para mejorarlo a diario.

Está en constante actualización y crecimiento al ser el gestor de contenidos más usado del mundo, se actualiza constantemente para introducir mejoras, corregir vulnerabilidades y añadir prestaciones.

De esta manera y conociendo las vulnerabilidades obtenida en las pruebas de penetración, escaneo con la suite de Kali Linux y teniendo en cuenta la actualización de los certificados de seguridad permiten tener un portal web robusto que mitigue gran parte de las vulnerabilidades encontradas.

# 12 DIVULGACION

Como parte esencial de la gestión de resultados en las pruebas **PENTESTING PARA EL PORTAL WEB DE LA USPEC, APOYADO EN EL PROYECTO DE SEGURIDAD OWASP**, tendrá como fuente de difusión el repositorio institucional de la Universidad Nacional Abierta y a Distancia UNAD, permanecerá como referencia de consulta para las personas externas y estudiantes, además se brinda la capacitación al personal encargado en el área de tecnología para que se lleve a cabo el proceso de mitigar los riesgos.

# 13 CONCLUSIONES

Las organizaciones deben hacer uso de herramientas que les permitan detectar vulnerabilidades en sus portales web, esto debido al crecimiento del internet y a la gran cantidad de aplicaciones web que pueden vulnerar los sistemas de información al no contar con las instrumentos que permitan poseer estrategias actualizados de protección que garanticen ser menos propensos a ataques o amenazas que se puedan presentar.

Hay portales que cuentan con sistemas de desarrollo que están obsoletos, los cuales son vulnerables como sus formas de autenticación, bases de datos y son blancos fáciles de atacar. Relacionando lo anterior, mediante sus mecanismos pruebas de penetración y ser menos propensos a los ataques más como falsas peticiones, denegación de servicios, inyección de código, entre otros, que se pueden prevenir mediante las guías Project y políticas adecuadas que garanticen continuidad.

Con medidas de pruebas de penetración web, vulnerabilidades presentadas, impiden estar menos expuestos cuando se evalúa la seguridad de un portal, sin embargo es alta la probabilidad de eventos porque no se pueden precisar cuándo nos van a realizar un ataque y vulnerar la integridad confidencialidad y disponibilidad de la información.

Mediante pruebas de penetración es posible analizar las debilidades con las que cuenta el portal web de la Unidad de Servicios Carcelarios y Penitenciarios USPEC, de forma que se permita aumentar su seguridad cumpliendo con las bases del proyecto OWASP para garantizar sistemas más seguros permitiendo así poder cubrir vulnerabilidades existentes.

Los ataques a portales web son más comunes cada vez, con el fin del lograr explotar vulnerabilidades y mirar cómo se encuentra un sistema si es seguro utilizando las herramientas que suministra Kali Linux, las cuales pueden ser herramientas de código abierto como comerciales que permiten hallar soluciones a distintos sitios apoyado del proyecto OWASP.

Las técnicas de pruebas de penetración son proyectos que fortalecen los portales web, que se examinen como el de Unidad de Servicios Carcelarios y Penitenciarios mediante la seguridad Informática y sus herramientas de aplicaciones web, herramientas de vulnerabilidades y escaneos los cuales permiten detectar factores de riesgos y contribuir de la mejor manera el proyecto OWAS Actualizando y mitigando ciertos afectados de seguridad con la habilidad del investigador.

Es un error creer que las organizaciones por más mínimos movimientos que generen no puede ser blanco de un ataque por eso se deben tener información de los ataques a los que están expuestos a diario las organizaciones y en este caso documentarse y generar píldoras que alerten la unidad de servicios carcelarios y penitenciarios USPEC de los que están expuestos.

## BIBLIOGRAFÍA


ALONSO CEBRIÁN, José María. GUZMÁN SACRISTÁN, Antonio. LAGUNA DURÁN, Pedro. MARTÍN BAILÓN, Alejandro. Ataques a aplicaciones web [en línea], [Consultado el 25 de noviembre de 2017]. Disponible en Internet: https://www.exabyteinformatica.com/uoc/Informatica/Seguridad_en_bases_de_datos/Seguridad_en_bases_de_datos_(Modulo_2).pdf.

AMAYA TARAZONA, Carlos Alberto. Unidad 1 Seguridad en Aplicaciones Web. Internet: datateca UNAD UNIDAD1_SEGURIDAD_EN_APLICACIONES_WEB_2014.pdf.

ARANGO QUINTERO, Juan Carlos. Tema 6. Diseño metodológico [en línea], 7 de diciembre 2014 [3 de agosto del 2017]. Disponible en internet: https://dokumen.tips/education/tema-6-diseno-metodologico.html.

ASCENCIO MENDOZA, Martha. MORENO PATIÑO, Pedro Julián. Desarrollo de una Propuesta Metodológica para Determinar la Seguridad en una Aplicación Web [en línea], año 2011 [5 de octubre de 2017]. Disponible en internet: http://repositorio.utp.edu.co/dspace/bitstream/11059/2511/1/0058A811.pdf.

Expresión Binaria. Prueba de Intrusión sobre aplicaciones [en línea], 20 de febrero 2011 [19 de agosto de 2017]. Disponible en internet: http://www.expresionbinaria.com/pruebas-de-intrusion-sobre-aplicaciones/

HERNÁNDEZ SAUCEDO, Ana Laura. MEJÍA MIRANDA, Jezreel. Guía de ataques, vulnerabilidades, técnicas y herramientas para aplicaciones Web [en linea], [17 de agosto de 2017]. Disponible en internet: http://recibe.cucei.udg.mx/revista/es/vol4-no1/computacion05.html.

MARTÍNEZ, John. GIRALDO, Andrés. Auditoria de Seguridad Informática [en linea], [20 de septiembre de 2017]. Disponible en internet: http://artemisa.unicauca.edu.co/~ecaldon/docs/audit/ponencia_PASSWORD_siti2004.pdf.

CABALLERO QUEZADA, Alonso Eduardo. Pruebas de penetración contra aplicación web [en linea], [17 de octubre de 2017]. Disponible en internet: http://www.reydes.com/archivos/slides/eventos/T_PdPAW_Alonso_ReYDeS.pdf

PALANCO, José Ramón. W3af un framework de test de intrusión web [en linea] 21 noviembre 2008 [21 de septiembre de 2017]. Disponible en internet: https://www.owasp.org/images/b/b5/W3af_owasp_spain_iv.pdf.

RODRÍGUEZ, Ricardo Martín. Algunos ejemplos y defensas contra el clickjacking [en línea], 2013 [22 de noviembre de 2017]. Disponible en internet: http://blog.elevenpaths.com/2013/10/algunos-ejemplos-y-defensas-contra-el.html

The OWASP foundation. OWASP top 10-2013 Los 10 más críticos en aplicaciones Web [en línea], 2013 [5 de febrero del 2017]. Disponible en internet: https://www.owasp.org/images/5/5f/OWASP_Top_10_2013_Final_Espa%C3%B1ol.pdf.

The OWASP foundation. OWASP Zed Attack Proxy Project. [En línea], 2013 [10 de septiembre del 2017]. Disponible en internet: https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project

UNAD. Lección 26 test de penetración, [17 de agosto de 2015] Disponible en internet: http://datateca.unad.edu.co/contenidos/233016/EXE_SAM/leccin_26_test_de_penetracin.html.

Universidad del Atlántico. Diseño Metodológico Preliminar [en línea], [20 de agosto del 2017]. Disponible en internet: http://www.uniatlantico.edu.co/uatlantico/sites/default/files/docencia/facultades/pdf/ciencias-juridicas/guia%20_monografia_diseno_metodologico.pdf

COLOMBIA. Congreso de Colombia. Ley 1273 de 05 enero 2009. Delitos Informáticos. Archivo General de la Nación. Bogotá 5 de enero 2009, 1273. Pagina1 – Pagina 4.

ISO 27000. Sistemadegestióndeseguridaddelainformación [en línea], 2014[15 de noviembredel2017]. Disponible en internet:http://www.iso27000.es/download/doc_sgsi_all.pdf

QUINTERO, José Luis.ANALISISYGESTIÓNDELRIESGO [En línea], [15 de noviembredel2017]. Disponible en internet: http://www.aec.es/c/document_library/get_file?uuid=b3945e58-17f2-4dc0-88ac-863ae9f998cb&groupId=10128.

# ANEXOS

## Anexo 1. Resumen analítico especializado R.A.E

| | |
|---|---|
| **1. TEMA** | Seguridad Informática |
| **2. TÍTULO** | Pentesting para el portal web de la USPEC, apoyado en el proyecto de seguridad OWASP |
| **3. AUTORES** | Jhonnier Yesid Zúñiga Mosquera |
| **4. FUENTESBIBLIOGRÁFICAS** | |


ALONSO CEBRIÁN, José María. GUZMÁN SACRISTÁN, Antonio. LAGUNA DURÁN, Pedro. MARTÍN BAILÓN, Alejandro. Ataques a aplicaciones web [en línea], [Consultado el 25 de noviembre de 2017]. Disponible en Internet: https://www.exabyteinformatica.com/uoc/Informatica/Seguridad_en_bases_de_datos/Seguridad_en_bases_de_datos_(Modulo_2).pdf

AMAYA TARAZONA, Carlos Alberto. Unidad 1 Seguridad en Aplicaciones Web. Internet: datateca UNAD UNIDAD1_SEGURIDAD_EN_APLICACIONES_WEB_2014.pdf.
ARANGO QUINTERO, Juan Carlos. Tema 6. Diseño metodológico [en línea], 7 de diciembre 2014 [3 de agosto del 2017]. Disponible en internet: https://dokumen.tips/education/tema-6-diseno-metodologico.html.

ASCENCIO MENDOZA, Martha. MORENO PATIÑO, Pedro Julián. Desarrollo de una Propuesta Metodológica para Determinar la Seguridad en una Aplicación Web [en línea], año 2011 [5 de octubre de 2017]. Disponible en internet: http://repositorio.utp.edu.co/dspace/bitstream/11059/2511/1/0058A811.pdf.
Expresión Binaria. Prueba de Intrusión sobre aplicaciones [en línea], 20 de febrero 2011 [19 de agosto de 2017]. Disponible en internet: http://www.expresionbinaria.com/pruebas-de-intrusion-sobre-aplicaciones/

HERNÁNDEZ SAUCEDO, Ana Laura. MEJÍA MIRANDA, Jezreel. Guía de ataques, vulnerabilidades, técnicas y herramientas para aplicaciones Web [en linea], [17 de agosto de 2017]. Disponible en internet: http://recibe.cucei.udg.mx/revista/es/vol4-no1/computacion05.html.
MARTÍNEZ, John. GIRALDO, Andrés. Auditoria de Seguridad Informática [en linea], [20 de septiembre de 2017]. Disponible en internet: http://artemisa.unicauca.edu.co/~ecaldon/docs/audit/ponencia_PASSWORD_siti2004.pdf.

CABALLERO QUEZADA, Alonso Eduardo. Pruebas de penetración contra aplicación web [en linea], [17 de octubre de 2017]. Disponible en internet: http://www.reydes.com/archivos/slides/eventos/T_PdPAW_Alonso_ReYDeS.pdf

| | |
|---|---|
| **5. AÑO** | 2018 |
| **6. RESUMEN** | El presente trabajo es el resultado de las pruebas de penetración realizadas al portal web USPEC basados en el proyecto OWASP utilizando herramientas de escaneos de vulnerabilidades que demuestran que el sitio web no es seguro porque posee un sistema de contenido obsoleto el cual no es compatible para actualizar a la versión más reciente del sistema de contenidos generando así el desarrollo de un nuevo portal web y corrigiendo las vulnerabilidades contando con la seguridad adecuada de acuerdo al top 10 de vulnerabilidades y a las guía de recomendación del proyecto OWASP, como actualización del sistema de contenido a Word press certificados de seguridad doble autenticación para administración del sitio entre otras. |
| **7. PALABRASCLAVES** | Pentesting, Portal web USPEC, Seguridad informática, OWASP, Vulnerabilidad Web, Ataques. |
| **8. CONTENIDOS** | La necesidad de contar con un portal web seguro que hacen parte de la consulta y publicación de información día a día, esto genera que se cuente con un portal web seguro, que garantice la confidencialidad, disponibilidad e integridad de la información los servicios prestados. |
| **9. DESCRIPCION DEL PROBLEMA** | |

Toda organización, empresa o entidad formalizada y estructurada posee una información que la define y que fundamenta todas sus operaciones diarias y esenciales. Dependiendo de la función que desempeña dicha entidad o empresa en la sociedad se define la característica de la información y el nivel de protección que se requiere para la misma. En la Unidad de Servicios Penitenciarios Carcelarios de Colombia, USPEC, entidad que se dedica al manejo jurídico y administrativo del sistema penitenciario y carcelario a nivel nacional, la información que se maneja no sólo es de carácter autentico y legítimo y se encuentra por ley bajo protección especial del Estado Colombiano ya que se trata de todo lo que concierne a la Población Privada de La Libertad y a la salvaguarda del Sistema Carcelario del País. Siendo el portal WEB el medio de acceso por excelencia a la información de la Unidad, éste requiere de un sistema de acompañamiento que vele por su seguridad, integridad y disponibilidad en el acceso, manejo y manipulación de la información. No obstante, el gestor donde se aloja la información que actualmente maneja la Unidad de Servicios Penitenciarios y Carcelarios, no cumple cms o gestor de contenidos con los estándares de seguridad que consienta a los usuarios y al administrador acceder y descargar la información veraz y autentica de la Unidad.

Es importante mencionar que el portal web está diseñado en un sistema de gestión de contenido llamado Joomla donde todo lo que se publica queda guardado en una base datos conectado a este gestor y que en el momento no se encuentra actualizado. El portal web de la Unidad se encuentra en la versión 2.5 de Joomla y actualmente este gestor se encuentra en la versión 3.8, no ha sido posible su actualización porque tiene implementados Plugins que no dejan, ni permiten realizar esta operación, motivo por el cual la parte administrativa se encuentra preocupada, ya que no saben qué tan seguro es su portal.

A través de los conocimientos adquiridos en este posgrado se tratará de encontrar las vulnerabilidades posibles y las consecuencias que se puedan conllevar por esta problemática, y de esta forma buscar mitigar por medio de estrategias y mecanismos de seguridad este problema. Es necesario y fundamental para la Unidad de servicios Penitenciarios y Carcelarios del país contar con un portal WEB con estándares óptimos de seguridad, integridad y disponibilidad tanto para los usuarios finales como para los usuarios administradores en la Unidad de Servicio Penitenciarios y Carcelarios. Si examinamos la importancia de la seguridad en el portal WEB y en especial, en el contexto de que actualmente la unidad es la encargada del Sistema Carcelario Nacional del país, es un tema con un elevado grado de jerarquía, pues hoy dado los avances tecnológicos, los portales web no solo son la cara principal de la organización, sino el medio a través del cual se establecen canales de comunicación con usuarios que requieren de los servicios de una organización o establecer un contacto necesarios entre usuarios, proveedores o administradores de la misma. Es así, como se ha estimado la necesidad a través del presente trabajo, realizar escaneos y pruebas de penetración basada en metodologías de seguridad como el proyecto OWASP en el portal web de la Unidad de Servicios Penitenciarios y Carcelarios, donde se evidencia que es de fundamental utilidad para establecer posibles fallas, malversaciones o intromisiones que afecten la seguridad del sistema.

| | |
|---|---|
| **OBJETIVOS** | OBJETIVO GENERAL<br><br>Implementar medidas de seguridad mediante escaneos y pruebas de penetración basado en el proyecto OWASP para el portal WEB de la Unidad de Servicios Penitenciarios y Carcelarios USPEC con el fin de mitigar riesgos a la seguridad de la información del mismo.<br><br>OBJETIVOS ESPECÍFICOS<br><br>✓ Identificar las debilidades de seguridad del portal web de la unidad de servicios carcelarios y penitenciarios USPEC, mediante técnicas de penetración para fortalecer soporte logístico y administrativos requeridos para la apropiada labor de los servicios penitenciarios. Aplicando el proyecto de seguridad de OWASP.<br><br>✓ Realizar el levantamiento de la información, estado del portal, normas, mostrando procesos de identificación, rastreo y diagnóstico de los problemas de seguridad del portal web.<br><br>✓ Documentar y realizar las pruebas de penetración para el portal web de la unidad de servicios penitenciarios la USPEC.<br><br>✓ Implementar una solución para mitigar las vulnerabilidades diagnosticadas a través del proyecto OWASP en el actual portal web de la USPEC “Unidad de Servicios penitenciarios y carcelarios del país” |

| | |
|---|---|
| **METODOLOGÍA** | Este proyecto presenta una metodología para gestionar la auditoria de seguridad del portal web de la unidad penitenciaria y carcelaria USPEC basado en el proyecto OWASP que identifica los principales riesgos de seguridad en aplicaciones y ofrece soluciones para las mejores prácticas. El portal web por su contenido informativo además de manejar proyectos, y publicaciones de los servicios a los requerimientos y/o servicios de bienes y financieros para las reclusiones a nivel nacional como para usuarios internos y externos de la unidad. Se facilita que al poseer un portal web desactualizado comiencen a ocurrir errores y desviaciones que pueden comprometer la propia subsistencia del portal, Esta realidad causa la necesidad de realizar procesos de auditoría del funcionamiento que permitan detectar problemas graves de vulnerabilidad, establecer políticas, realizar actualización de su sistema de contenidos e incluso detectar problemas de programación que logran poner en riesgo la operación futura del portal web USPEC.<br><br>Durante los procesos de auditoría Pent Test es trascendental, garantizar que las consecuencias sean correctas y completas, para obtener un resultado uniforme, reduciendo la importancia de los niveles de técnica, instrucción, audacia, conocimiento del portal web auditado. La metodología para la detección de vulnerabilidades en el portal web mediante la las herramientas de la Suite de Kali Linux expuestas en este proyecto PENTESTING PARA EL PORTAL WEB DE LA USPEC, APOYADO EN EL PROYECTO DE SEGURIDAD OWASP, Consta de cuatro fases probadas con las herramientas del software libre Kali, mediante las cuales se busca obtener las vulnerabilidades sobre el portal web USPEC. Esta metodología se soporta cada etapa en herramientas software mediante el proyecto OWASP.<br><br>▪ Reconocimiento y Definiciones.<br>▪ Servicios técnicos y Escaneo de puertos<br>▪ Revisiones de las configuraciones.<br>▪ Resultados y Recomendaciones |

| | |
|---|---|
| **PRINCIPALESREFERENTES TEÓRICOS** | Marco Teórico, Antecedentes, Marco referencial. |
| **PRINCIPALESREFERENTES CONCEPTUALES** | Seguridad de la información, pruebas de vulnerabilidad. |
| **RESULTADOS** | En base a la demostración de las fallas de seguridad web del portal web de la Uspec se tuvo como resultado la realización de un portal web con un sistema de contenido robusto que cuenta con certificados de seguridad actualizados, además seguridad para la administración con doble autenticación. |
| **CONCLUSIONES** | A lo largo de este proyecto de grado aplicado se ha demostrado que es de vital importancia que los portales web cuenten con la seguridad adecuada. Ningún sistema de seguridad es infalible y, esto quiere decir que en las aéreas de tecnología en sus Data Center alojamientos de servidores los dispositivos como firewalls y dispositivos de seguridad, son importantes en su implementación siempre y cuando vayan acompañados de los manuales que le permitan tener un sistema robusto de ataques PENTEST. La probable solución posible es la realización de auditorías periódicas y la creación de una cultura de la seguridad para concienciar a todos los funcionarios de los riesgos a los que se exponen. |